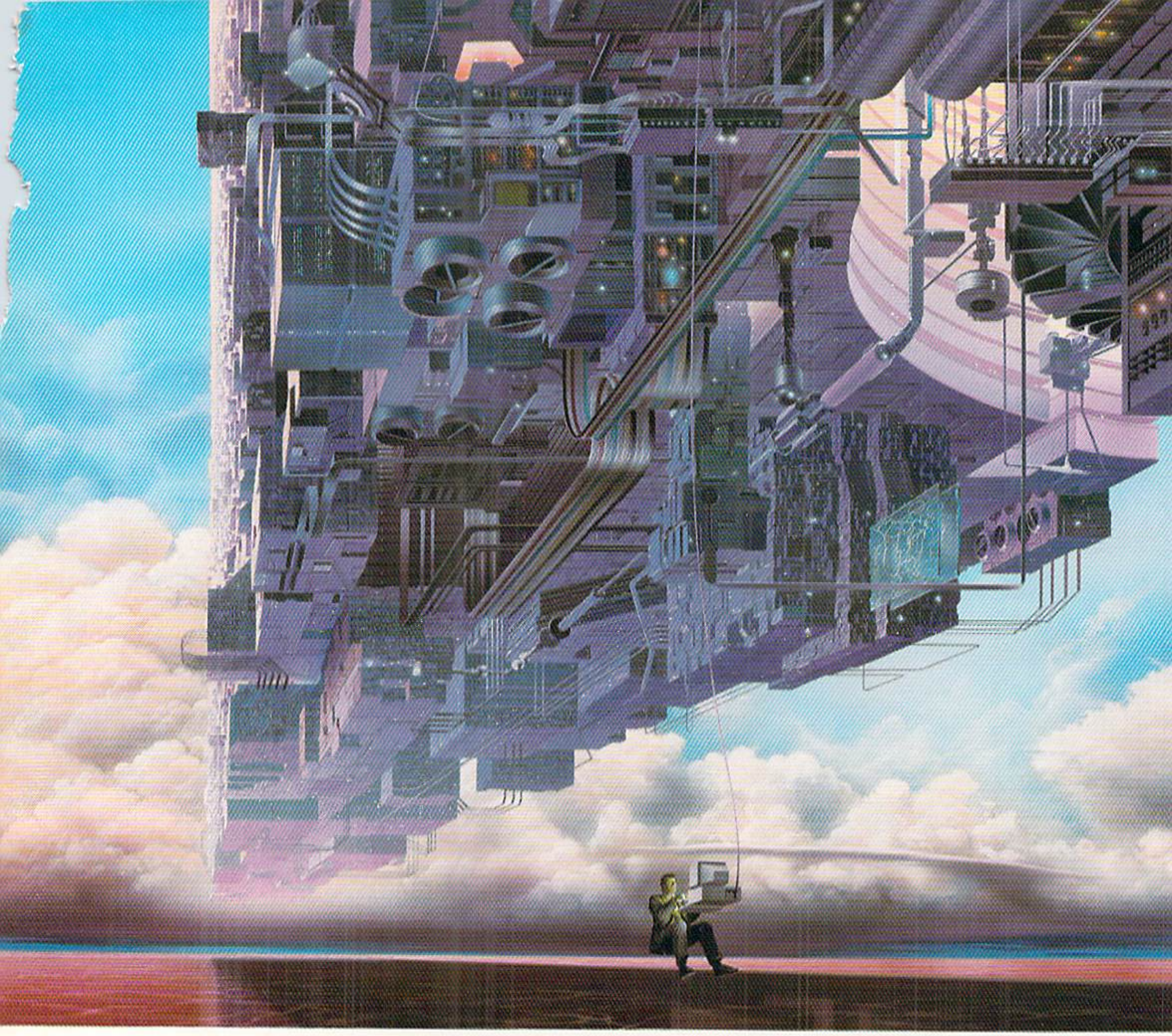
COMMODORE USERS

Ahoy!

$2.95/ CAN. $3.35
JUNE 1988





Ahoy!

CONTENTS

DEPARTMENTS

FEATURES

*Includes program: *Directory Scanner* for the C-64 and C-128
**Includes program: *Lost Dutchman's Mine* for the 64, 128, Plus/4

PROGRAMS

Cover art by James Regan; photography by Steven Hunt/The Image Bank and Michael R. Davila

President
Michael Schneider

Publisher/Editor
David Allikas

Art and Production Director
Laura Pallini

Managing Editor
Michael R. Davila

Senior Editor
Cleveland M. Blakemore

Technical Editors
David Barron
Bob Lloret

Consulting Editors
Morton Kevelson
Tim Moriarty
Dale Rupert

Entertainment Editor
Arnie Katz

Art Production
Christopher Carter

Circulation Director
W. Charles Squires

Advertising Director
Lynne Dominick

Director of Promotion
Mark Kammerer

Promotion Art Director
Stacy Weinberg

Controller
Dan Tunick

Advertising Representative
JE Publishers' Representative
6855 Santa Monica Blvd., Suite 200
Los Angeles, CA 90038
(213) 467-2266
Boston (617) 437-7628
Dallas (214) 660-2253
New York (212) 724-7767
Chicago (312) 445-2489
Denver (303) 595-4331
San Francisco (415) 864-3252

ISSUE NO. 54 JUNE 1988

Ahoy! (ISSN #8750-4383) is published monthly by Ion International Inc., 45 W. 34th St., Suite 500, New York, NY 10001. Subscription rate: 12 issues for $23.00, 24 issues for $44.00 (Canada $30.00 and $55.00 respectively). Second class postage paid at New York, NY 10001 and additional mailing offices.

Postmaster, send address changes to *Ahoy!*, 45 W. 34th Street, Suite 500, New York, NY 10001. Direct all address changes or matters concerning your subscription to *Ahoy!*, P.O. Box #341, Mt. Morris, IL 61054 (phone: 815-734-4151). All editorial inquiries and products for review should be sent to *Ahoy!*, 45 W. 34th St., Suite 500, New York, NY 10001.

LET'S FACE IT—MAIL ORDER HOUSES ARE *NOT* ALL THE SAME!

SOFTWARE SUPPORT INTERNATIONAL

CONVENIENCE—FAST SERVICE—RELIABILITY—SUPPORT

Hands On Software, Inc. is now distributing its own products. "What does that mean to me," you say! Well, for one thing, we have expanded our product line to include games and utilities from other publishers, accessories, Commodore authorized service, and even a selection of pre-owned titles. The technical staff here is second to none, and is well-known to thousands of Commodore users. We are committed to your satisfaction—and are prepared to prove it!

ACCESSORIES

1541/71 Serial or Power Cable	4.95
C-64 Color Monitor Cord	4.95
C-64 Repairable Power Supply	39.95
RS-232 Interface	36.95
MW 350 Printer Interface	49.95
Aprospand 64	27.95
Aprospand Extender Cable	15.95
12" Printer Stand/Wire	8.95
Commodore 1351 Mouse	37.95
Mouse Mat	8.95
41/71/64/128 Dust Cover	each 8.95
Disk Notcher—Square Cut	4.95
5.25" Disk Drive Cleaner	5.95
3.5" Disk Drive Cleaner	6.95
10 Cnt. Disk Storage	1.95
50 Cnt. Disk Storage	8.95
100 Cnt. Disk Storage w/Lock	12.95
3.5" Disk Labels—25 Cnt.	1.00
5.25" Disk Labels—96 Cnt.	1.00
3.5" DS/DD Diskettes	1.29
5.25" DS/DD Diskettes—Black	.49
5.25" DS/DD Diskettes—12 Colors	.79
Write Protects—100 Cnt. Black	1.00
Tyvek Sleeves—High Quality	each .09
Slik Stik—Joystick	6.95
Quick Shot II—Joystick	7.95
Tac 2—Joystick	10.95
Tac 3—Joystick	11.95

Commodore Authorized Service by Russell & Associates

Repairs: Commodore	
C-64 Keyboard	54.95
C-128 Keyboard	79.95
1541 Align	39.95
1541 Repair (w/align)	65.00
1571 Align	39.95
1571 Repair (w/align)	89.95
1702 Monitors	75.00
1902 Monitors	85.00
Commodore Printers	85.00

Service:	
C-128 Rom Upgrade	42.50
1571 Rom Upgrade	19.95
C-128 64K Ram Video (great for Basic 8)	19.95
Repairs: Amiga	
Amiga 500	135.00
Amiga 1000	135.00
Amiga 2000	135.00
A1010 Align	54.95
A1010 Repair	145.00

All C-64's & C-128's **must** have Power Supply included. Thirty-day warranty on all repairs. Buy a Power Supply from Software Support International and get a 60-day warranty. Please include a detailed description of your problem and try to isolate if possible.

Russell and Associates reserves the right to refuse any repair. Repair jobs will be charged $3.00 per piece for return shipping/handling.

Abacus Software

Basic Compiler 64	24.97
Basic Compiler 128	36.97
Cad Pak 64	24.97
Cad Pak 128	36.97
Chart Pak 64	24.97
Chart Pak 128	24.97
Geos Inside & Out Book	12.97
Geos Disk	9.97

ACCESS Software Incorporated

Echelon	24.97
LB & Tourn. & Exec. 1	12.97
Mach 128	30.97
Tenth Frame	24.97
World Class Leader Board	24.97
W/C L/B Fam/Cour 1 or 2	12.97

ACCOLADE

Ace of Aces	18.97
Apollo 18	18.97
Bubble	18.97
Card Sharks	18.97
Fourth & Inches	18.97
Hardball	18.97
Killed Until Dead	18.97
Miniputt	18.97
Plasmatron	9.97
Power at Sea	18.97
Project Space Station	9.97
Test Drive	18.97
The Train	18.97

ACTIONSOFT

Thunder Chopper	18.97
Up Periscope	18.97

ACTIVISION ENTERTAINMENT SOFTWARE

Aliens	21.97
Gee Bee Air Rally	18.97
Labyrinth	18.97
Last Ninja	21.97
Maniac Mansion	21.97
Might & Magic	24.97
Portal	24.97
Shanghai	18.97

Berkeley Softworks

Desk Pak I	10.98
Font Pak I	10.98
Geos 64	36.97
Geos 128	42.97
Geocalc 64	30.97
Geocalc 128	42.97
Geodex 64	10.98
Geodex 128	42.97

Berkeley Softworks

Geofile 64	30.97
Geofile 128	42.97
Geoprint Cable	24.97
Geoprogrammer 64	42.97
Geoprogrammer 128	Call
Geopublish 64	42.97
Geopublish 128	Call
Geospell	18.97
Geowriter's Workshop 64	30.97
Geowriter's Workshop 128	42.97

Broderbund Software

Carmen Sandiego USA	24.97
Carmen Sandiego World	21.97
Cauldron 1 & 2	18.97
Magnetron	17.16
Print Shop	27.97
Print Shop Companion	21.97
P/S Graphics 1 or 2 or 3	17.67
P/S Holiday Graphics	15.97
Superbike Challenge	13.92

CAPCOM

Ghosts & Goblins	18.97
Side Arms	18.97
1942	18.97

Breakthru	20.40
Commando	13.92
Kid Niki	20.40
Speed Buggy	18.46

DATA SOFT

Alternate Reality: City	18.97
Alternate Reality: Dungeon	28.81
Road Wars	Call
Tobruk	Call
Video Title Companion 1 or 2	12.97
Video Title Shop	18.97

Dictionary Disk	9.97
Digital Superpak 2	60.97
Pocket Filer 2	36.97
Pocket Planner 2	36.97
Pocket Writer 2	36.97

ELECTRONIC ARTS

Alien Fires	Call
Amnesia	27.36
Bard's Tale 1 or 2	27.36
Battle Droids	Call
Chessmaster 2000	24.97
Chuck Yeager's AFS	24.06
Death Lord	Call
Delta Patrol	14.16
Demon Stalker	18.97

ELECTRONIC ARTS

Dragon's Lair	17.46
Earth Orbit Station	22.06
Hunt For Red October	22.78
Instant Music	20.76
Legacy of the Ancients	20.76
Marble Madness	20.76
Master Ninja	Call
Mavis Beacon Typing	Call
Monopoly	27.40
Outrageous Pages	33.96
Paper Clip 3	33.96
Paperclip Publisher	Call
Pegasus	22.06
Pegasus Scenerio 1	Call
Rockford	20.80
Sanxion	14.16
Scrupples	24.10
Skate or Die	20.76
Skyfox II	20.76
Strike Fleet	18.97
Twilights Ransom	Call
World Tour Golf	20.76

EPYX

California Games	24.97
Champ. Wrestling	24.97
Create A Calendar	18.97
Death Sword	Call
Destroyer	24.97
Four X Four Racing	Call
The Games: Winter Edition	Call
Graphics Scrap 1 or 2 or 3	15.97
Home Video Producer	Call
Impossible Mission II	Call
L.A. Crackdown	Call
Omnicron Conspiracy	24.97
Sporting News Baseball	Call
Spy vs. Spy 3	15.97
Street Sports: Baseball	24.97
Street Sports: Basketball	24.97
Street Sports: Soccer	Call
Sub Battle Simulator	24.97
Summer Games II	13.92
Winter Games	13.92
World Games	24.97

Firebird

Guild of Thieves	24.97
Knight Orc	15.97
Revs +	15.97
Star Glider	24.97
Talking Teacher	21.97
The Pawn	24.97
The Sentry	24.97
Tracker	24.97

GAMESTAR

Champ. Baseball	18.97
Champ. Basketball	21.97
GFL Football	21.97
Two on Two Basketball	18.97

MICRO PROSE

Airborne Ranger	21.97
F-15 Strike Eagle	21.97
Gunship	21.97
Pirates	24.97
Project Stealth Fighter	24.97
Red Storm Rising	24.97
Silent Service	21.97

Bad Street Brawler	Call
Bop & Rumble	18.97
Bop & Wrestle	18.97
Crossword Magic	30.97
Defender of the Crown	21.97
Deja Vu	21.97
Gauntlet	21.97
Harrior Combat Simulator	Call
Head Over Heels	12.97
High Roller	18.97
Impact	Call
Indoor Sports	18.97
Indy Jones/Temple of Doom	Call
Infiltrator	18.97
Infiltrator II	21.97
Into the Eagle's Nest	18.97
King of Chicago	Call
Mutants	12.97
Paperboy	21.97
Road Runner	Call
SuperStar Ice Hockey	21.97
SuperStar Misl Soccer	Call
SuperStar Soccer	21.97
Taipan	12.97
Three Stooges	21.97
Warp Speed 64/128 40/80	30.97
Winter Challenge	12.97
Wizball	12.97
Xevious	18.97

ORIGIN

Autoduel	33.36
Moebius	26.88
Ogre	20.40
Ultima I	26.88
Ultima 3	26.88
Ultima 4	39.84
Ultima 5	Call

Professional Software Inc.

Fleet Filer 64/128	24.97
Fleet System II +	36.97
Fleet System III 128	42.97
Fleet System IV	48.97

SPRINGBOARD

Certificate Maker	26.88
Certificate Maker Libr. I	20.40
Newsroom	33.36
N/R Clipart 1 or 3	20.40
N/R Clipart 2	26.88
P/S Graphic Expander	23.64

subLOGIC

Flight Simulator II	33.97
Flight Sim Scenery 1-6	14.16 ea.
Flight Sim Scenery 7	17.46
Jet	27.36
Stealth Mission	33.96

TIMEWORKS

Data Manager 2	17.16
Data Manager 128	33.36
Desktop Publisher	Call
General Ledger	33.36
Partner 64	33.36
Partner 128	39.84
Swiftcalc/Sideways 64	17.16
Swiftcalc/Sideways 128	33.36
Sylvia Porter F/P 64	33.36
Sylvia Porter F/P 128	46.32
Word Writer 3	33.36
Word Writer 128	33.36

UNISON WORLD INCORPORATED

Art Gallery 1 or 2	15.97
Art Gallery American Hist.	Call
Art Gallery Fantasy	15.54
Printmaster Plus	21.97

MISCELLANEOUS

Basic 8	29.95
Basic 8 Guide/Beginners	19.95
Bob's Pro Term 64	30.97
Bob's Pro Term 128	42.97
B/W Business Form Shop	24.97
CSM Drive Align Kit	30.12
CSM Protection Manual I	22.55
CSM Protection Manual II	26.15
Dark Castle (3-60)	21.97
Doodle	24.97
Font Master II	30.97
Font Master 128	36.97
Hes Mon 64 (cart)	9.95
Inside Commodore DOS	19.95
JK Lasser Money Manager	46.32
Merlin Assembler 64	30.97
Merlin Assembler 128	42.97
Microlawyer	36.97
Microleague WWF Wrestling	24.95
Personal Newsletter	Call
Strider's Classical Music—All 10 disks in Filer	49.95
Strip Poker	18.97
Poker Data Disks 1 or 2 or 3	12.97
Superbase 64	48.97
Superbase 128	48.97
Superbase the Book	10.57
Super Sunday	23.64
Superscript 64	30.97
Superscript 128	42.97

MANY OTHER TITLES AVAILABLE — CALL FOR PRICES!

Here they are! The products that have made us one of the most respected companies in the mail order business. Our utilities have become a staple in the libraries of thousands upon thousands of Commodore users worldwide. Original ideas and workmanship throughout, (a rarity nowadays!) our products are guaranteed to please. If you enjoy dealing with people who genuinely have your best interest at heart, look no further. Our products are created with you in mind. We offer programs that are protection free and most can be transferred to the 1581 disk drive. Also, we offer the best technical support in the business. Les and Dan are available five days a week to help with problems. If you see a program that interests you on this page, and you need more detailed information to make an educated decision, just write or call for our free catalog. Remember: Our Business is SOFTWARE SUPPORT!

Renegade

ALL NEW

THE CURE FOR RANDOM EXCESS

Random chance. Buy enough utilities, and every so often you'll get a good one, right? Well, that's one way to do it. One very expensive way. We've got a better idea!

We've had the hottest archival programmers in the industry develop a new series of utilities of startling power. They took the best ideas from the past and re-created them, using state of the art techniques combined with a conceptual grasp of the future of computing.

The results of this special project are now available in a single, devastating package. A next-generation package called RENEGADE.

FEATURES INCLUDE:

- Single or Dual High Speed Data Copier
- Single or Dual State of the Art Nibbler
- Directory Editor: Organize your Disks
- GCR Editor: For the Experienced Hacker
- Geos™ Modual: PARAMETERS and TOOLS
- 200 Parm: Oldies as well as our Latest
- Parameter Updates Released Every Two Months
- Copier Updates Released Every Four Months
- Compatible with C-64/128 with 1541/71 Drives
- Single or Dual Ultra Fast File Copier
- Capable of Archiving RapidLok Protection
- Scrolling M/L Monitor with Drive Mon
- Error Scanner with Unique Sector Editor
- Byte Pattern Scanner: High Speed Searches
- Upgradable Sub Menu: New tools in the works
- New Parameter updates only $9.95 each
- Major RENEGADE upgrades only $9.95 each
- Technical support available: absolutely.

Remember the first programs that came out when the C-64 was still a fairly new machine? Have you seen the newest software lately? You'd think they were running on a different computer. RENEGADE represents the same quantum leap forward in the utilities arena.

So if you're tired of relying on random chance to provide you with good utilities, get RENEGADE—next generation software, available today!

ONLY $34.95

C-128 Helper

Helper is an exciting new type of Software for the C-128 Computer with 80 column monitor.

- Screen Based Databasing
- User Definable
- Editor allows modifications
- Demo included that serves as a Basic 7.0 Electronic reference manual
- Screen dumps to printer
- 1581 version available

ONLY $24.95

Graphic Art Disks

Top quality graphics are hard to find. These will definitely please you.

- 10 Volumes available
- Themes followed in most
- Printmaster™/Printshop™ Side A compatible
- Easily transferred to 1581

ONLY $14.95 per Volume

Elite V3.0

Elite V3 copies the "impossible."

- Copies V2.0 of a popular 64/128 planner, filer (64 only), writer.
- Leaves the Backup protection free.

Why pay for Backups of titles you already own?

ONLY $6.95

Elite V4.0

Our Elite Series is second to none for its copying capabilities.

- Elite 4 created to copy titles protected to the MAX!
- Special MAX Copier included
- Update policy included
- Free Bonus—The Shotgun I

ONLY $14.95

Super Cat

The greatest disk library cataloger ever produced for the Commodore 64.

- Catalogs up to 640 Disks w/5000 titles per catalog
- Twice as many with 2 drives
- Load, save, edit, change any entry
- Prints a variety of reports and even labels
- Lightning fast and easy to use!

ONLY $24.95

Graphic Label Wizard

This is the ultimate label utility.

- Databasing with graphics included
- Supports Commodore as well as Epson compatible printers
- Printmaster™ Printshop™ Software Solutions™ compatible
- 1581 transferable
- Satisfaction guaranteed!

ONLY $24.95

C-128 Cannon

The total copy/utilities package created just for C128 owners.

- Nibblers—Single or Dual
- Fast Copiers—Single or Dual
- File Copiers—Single or Dual
- Scanners—Error and Density
- 1581 Utilities
- Directory editing
- 100 Free Kracker Jax Parms
- 1541/71/81 Drives Supported—Single or Dual

ONLY $34.95

Kracker Jax Revealed I & II

Our knowledge of protection schemes has made us famous. Now find out how we do what we do best. Our books are your key to REAL knowledge. OTHERS ONLY GIVE SURFACE INFO—we dig deep—much deeper. We can show you exactly how today's copy protection prevents you from backing up your software.

Revealed Book I with Reset Button
$23.50

Revealed II with Hesmon Cartridge
$23.50

Kracker Jax

THE REVOLUTIONARY ARCHIVAL SYSTEM!

Kracker Jax is the powerful parameter-based copying system that has taken the country by storm! What IS a parameter? It's a custom program that allows your 1541 or 1571 disk drive to strip ALL copy protection from your expensive software, leaving you with UNPROTECTED, TOTALLY BROKEN BACKUPS that can even be copied with a simple fast copier!

We declare **Kracker Jax** to be the best system of its kind on the market today! A bold claim? Maybe.

But don't take our word for it—if you want the REAL story on how good **Kracker Jax** is, just ask one of our customers. Don't worry. You won't have any problem finding one.

Vols 1-2-3-4 **Only $9.95 ea.**
Vols 5-6-7 **Still Only $19.95 ea.**

Hacker's Utility Kit

State of the art tools for the computerist ready to take control of his software library!

- Whole disk scanners
- GCR Editor
- Fast Data/Nibble copiers
- File tracer
- Byte Pattern Finder

Much, Much More—$19.95

Sysres™ Enhanced

The best—and we mean Best Basic enhancement system for the C-64.

- Adds over 25 major commands to Basic
- Extended Super DOS-Wedge
- Scrolling through Basic
- Renumber, Trace, Search, and many other features too numerous to mention
- 1541/71 fast loader included
- M/L monitor from Basic
- Transferable to 1581 Disk

ONLY $39.95

The best money you'll ever spend on your Commodore!

SOFTWARE SUPPORT INTERNATIONAL

Reader Service No. 137

ORDERING is simple: we accept money orders, certified checks, personal checks only on prior approval, VISA, MC, Discover and COD. Please add $3.00 shipping and handling per order. $2.00 additional for COD orders. Orders outside of North America please call for shipping charges. Defective items will be replaced at no charge. In stock orders are processed within 24 hrs. Returns are not accepted without authorization #. Washington Residents please add 7.3% sales tax. Software orders over $100.00 will be shipped UPS 2nd Day Air at no additional charge. U.S. orders only. All prices subject to change.

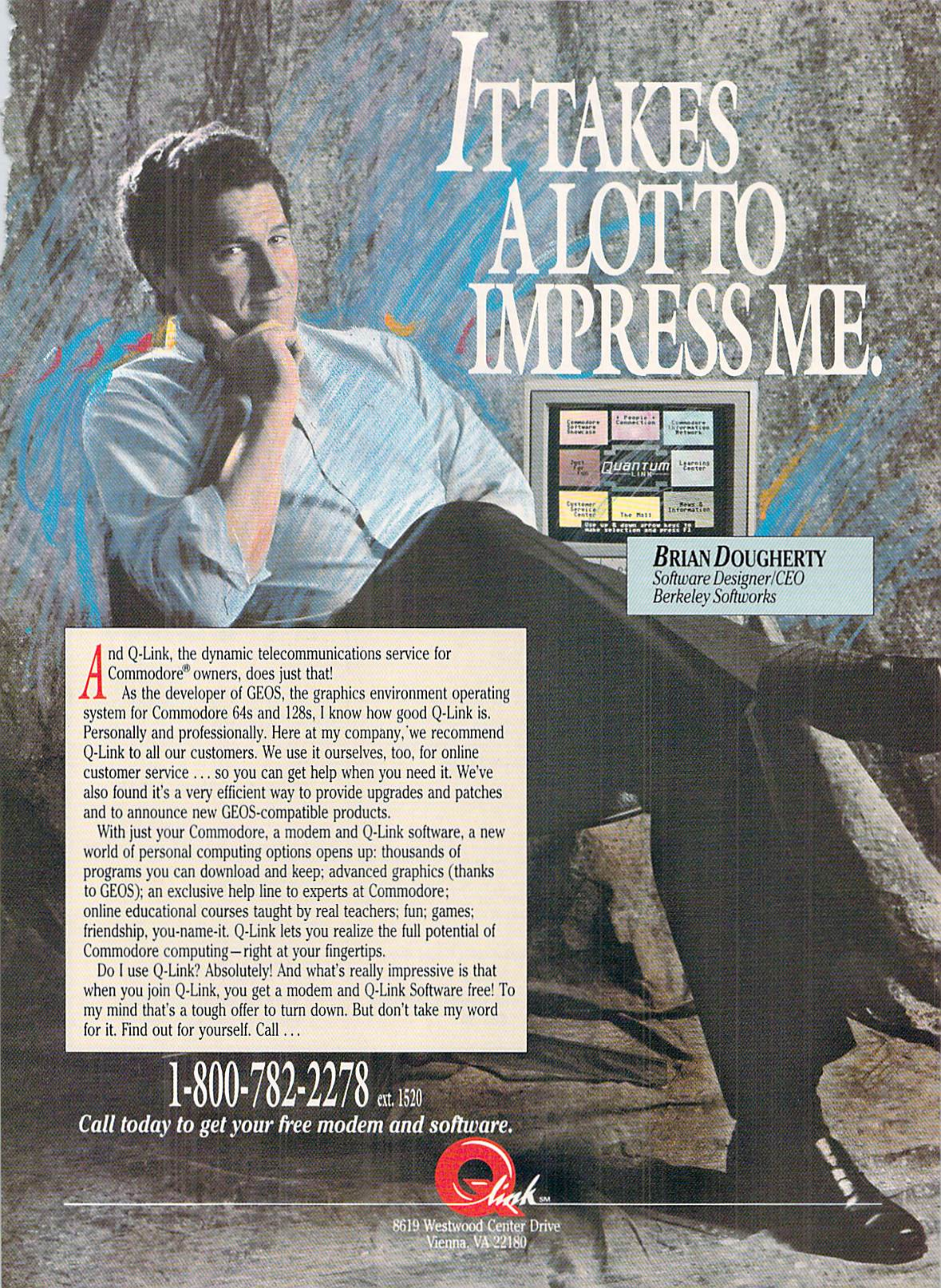
Program Submissions Invited
Need more info? Call or write for our free catalog.

Mail your order to: Software Support Int.-D13
2700 NE Andresen Road / Vancouver, WA 98661

Or call our toll-free order line at 1-800-356-1179, 9am-5pm Pacific time Monday-Friday.
After hours orders accepted at (206) 695-9648 7 days a week.
Technical support available. Call (206) 695-9648, 9am-5pm Pacific time, Monday-Friday.

VISA · MasterCard · DISCOVER

DEALERS — WE HAVE THE SUPPORT YOU'RE LOOKING FOR!


IT TAKES A LOT TO IMPRESS ME.
BRIAN DOUGHERTY
Software Designer/CEO
Berkeley Softworks
And Q-Link, the dynamic telecommunications service for Commodore® owners, does just that!
As the developer of GEOS, the graphics environment operating system for Commodore 64s and 128s, I know how good Q-Link is. Personally and professionally. Here at my company, we recommend Q-Link to all our customers. We use it ourselves, too, for online customer service . . . so you can get help when you need it. We've also found it's a very efficient way to provide upgrades and patches and to announce new GEOS-compatible products.
With just your Commodore, a modem and Q-Link software, a new world of personal computing options opens up: thousands of programs you can download and keep; advanced graphics (thanks to GEOS); an exclusive help line to experts at Commodore; online educational courses taught by real teachers; fun; games; friendship, you-name-it. Q-Link lets you realize the full potential of Commodore computing—right at your fingertips.
Do I use Q-Link? Absolutely! And what's really impressive is that when you join Q-Link, you get a modem and Q-Link Software free! To my mind that's a tough offer to turn down. But don't take my word for it. Find out for yourself. Call . . .
1-800-782-2278 ext. 1520
Call today to get your free modem and software.
Q-Link SM
8619 Westwood Center Drive
Vienna, VA 22180

VIEW FROM THE BRIDGE

In four and a half years of publishing *Ahoy!*, nothing has touched us as deeply as the outpouring of emotion from devoted readers these past few months. It was occasioned, of course, by our announced cutback in *Ahoy!* from 12 to 8 issues a year in order to make room in our schedule for *Ahoy!'s AmigaUser* the other 4 months. We anticipated that many 64 and 128 devotees would be unhappy with the change. We had no idea how many, or how unhappy! (A sampling is included in *Flotsam* on page 23.)

It makes us proud to know that those missing 4 issues make such a difference to so many readers. We wish that we could accommodate you with a return to monthly publication. Sadly, there aren't enough hours in the day to do so. And we'd rather publish 8 issues a year that meet our usual standards than rush 12 sub-par issues to press. Faced with the two alternatives, we know most of you must feel the same way.

On the bright side, you're holding a magazine containing 100% C-64 and 128 coverage, with no space devoted to an *Amiga Section* or Amiga news—and *with* first-rate features like the following:

• In April, Dale Rupert explained some of the more esoteric Commodore BASIC keywords. In this month's *Rupert Report* he does the same with the disk operating system, putting you *Right on Track* to some advanced drive commands. (Turn to page 32.)

• In March 1987, we began a new column intended to spread the word about the Commodore market's best-kept secret: the excellent COMAL programming language. Space considerations forced us to drop the column after the October issue, but it returns this month, penned as before by Richard Herring and offering as before almost as much useful information about programming in general as about COMAL in particular. (Turn to page 60.)

• A second column returns this month—sort of. From June '84 through March '86, we covered the subject of *Creating Your Own Games on the VIC and 64*—so completely, in fact, that we ran out of relevant information to offer. But so many fans of Cleve Blakemore's text adventures have asked for insights into his programming methods that he's going to go over some of the ground covered years back by Orson Scott Card. You longtime readers won't be bored—you'll find that Cleve's strategy for *Programming Your Own Text Games* is like no one else's. (Turn to page 39.)

• It had to happen! For the past three years our programmers have been devising ways to endow C-64's with C-128 capabilities. Now, they've turned their attention to the Amiga. If you envy Amiga owners their BOB capabilities, escape the limitations of your 64 with *MOB Blob.* (Turn to page 41.)

• For those who think that the Commodore 64's CPU is too slow to scroll text vertically, refer to Peter M.L. Lottrup's *Movie Scroll.* You'll be amazed to find how easily it can be done from BASIC. (Turn to page 30.)

• It's not because Cleve is now a staff member that he managed to write all four of the games included in this issue. We look at his programs with the same objectivity as we do everybody else's—and his contributions this month would have been easy sales even if they'd arrived by sea mail from Guatemala. *Phobia* is the latest "non-game" to spring from Cleve's mildly diseased mind. You must pass tests of both reflexes and reason to leave this madcap gameworld alive. (Turn to page 20.) In *Rashgar,* you're the laser-spewing title character who must guard the entrance to the X dimension against deadly Crystallions. (Turn to page 16.) *Snap Snake* stars the reptile of Texas legend in a hybrid BASIC-machine language arcade game. (Turn to page 43.) And *Lost Dutchman's Mine* gets you in the mindframe for the kind of text game programming described in this month's column. (Turn to page 39.)

We couldn't agree more with the devoted 8-bit users in our audience—there's a lot of life left in the 64 and 128. So stick with us. We've got a lot of living to do together!

—David Allikas

JUNE DISK BONUS

Mines of Merlin, the enhanced version of Cleve Blakemore's *Vault of Terror*, is included on this month's *Ahoy! Disk.* It features faster response time, an enhanced title screen, and is offered minus the bug that plagued the original. See page 21 for information on ordering the *Ahoy! Disk.*

SCUTTLEBUTT

SPECIAL EDUCATION AIDS • ADULT GAMES • GEOS WORD PROCESSOR • MUSIC SOFTWARE CATALOG • ONLINE BUSINESS INTERCHANGE • BIG BLUE READER UPDATE • GAMES FROM ACCOLADE, DATA EAST, FREE SPIRIT, SSI • ADVENTURE HINTS • TYPING PROGRAM • GEOS CONTEST

DATA INTERCHANGE

The CompuServe Electronic Data Interchange (EDI) Service has been established to provide businesses with a fast means of transmitting intercompany documents and information. Basic EDI services are enhanced by CompuServe services like database access, electronic mail, publishing and conferencing to create a total communications package.

CompuServe's packet data network provides local dial-up access to more than 85% of the United States, maintains uptime reliability in excess of 98%, and ensures error-free document and file transmission with the CompuServe EDI service. Document standards supported include the ANSI X12 standard, ANSI X12-based industry standard, and the entire TDCC family of document standards.

CompuServe, 614-457-8600 (see address list, page 14).

BECKERBASIC CONTEST

Abacus will award more than $25,000 in prizes to the users who write the best applications using *BeckerBASIC,* the GEOS programming aid. The grand prize is $1000; two second, third, and fourth prizes, and one hundred fifth prizes, consist of Abacus books and software in various combinations. Entries must be received by Abacus no later than August 31. For complete details, see the entry form inside packages of *BeckerBASIC*, or contact Abacus.

Abacus Software, 616-698-0330 (see address list, page 14).

NEW GRAPHICS GALLERIA

Inkwell Systems has reorganized its *Graphics Galleria* of *Flexidraw*-compatible clip art into 4-disk collections. Group I consists of Clip Art Potpourri, Borders & Signs, Maps of the World, and Holiday Themes. Group II contains Clip Art Potpourri II, Animals, The Office, and Futuristic Encounters. Each 4-disk group retails for $29.95; individual disks are $9.95.

Inkwell Systems, 619-268-8792 (see address list, page 14).

NEW BIG BLUE

Big Blue Reader 128/64, the third release of S.O.G.W.A.P. Software's Commodore to MS-DOS file transfer program, adds 1581 disk drive support. Two programs are included: *BBR 64,* which transfers Commodore DOS files to standard 5¼" MS-DOS format and vice versa using the 1571 drive; and *BBR 128,* which transfers Commodore DOS, MS-DOS, and C-128 CP/M files interchangeable on both 1571 and 1581 disk drives. Price is $45; upgrade cost from either of the previous versions is

$18 plus the original diskette.

S.O.G.W.A.P. Software, Inc., 219-724-3900 (see address list, page 14).

TYPING PROGRAM

Designed for students from Grade 4 up, *All The Right Type* covers all aspects of keyboarding, including correct posture and finger positioning. Speed and accuracy are built using real sentences and paragraphs rather than random sequences of letters. School version (with teacher's manual and backup) is $49.95; lab pack (disks and materials for five work stations, plus manual and backup) is $99.95; site license to make unlimited copies, $225.

Didatech Software, 604-299-4435 (see address list, page 14).

SIGNS & NUMBERS

Two educational programs from Free Spirit, each $19.95:

Easy Sign teaches the deaf hand sign alphabet by displaying the sign for a phrase or sentence typed in on the C-64.

The Number Show for the C-128 teaches numbers 1-9 to preschoolers and kindergarteners. Each part of the program is followed by a series of reinforcing exercises.

Free Spirit Software, Inc., 312-352-7323 (see address list, page 14).

SPECIAL ED AID

Designed to help teachers serve their special needs students, *Mindscape Methods for Mastery* ($19.95) matches over 40 Mindscape programs to 120 academic and social skills. The book includes lesson plans and ideas on supplementing the software with learning aids, demonstrating ways to integrate the programs into daily classroom lessons. A special section deals with the use of adaptive devices for handicapped learners.

Mindscape has also announced the availability of lab packs for nearly all their educational titles. Each pack contains five copies of the program disk, one manual, and one teacher guide, for twice the price of a single program.

Mindscape, Inc., 312-480-7667 (see address list, page 14).

ADULT GAMES

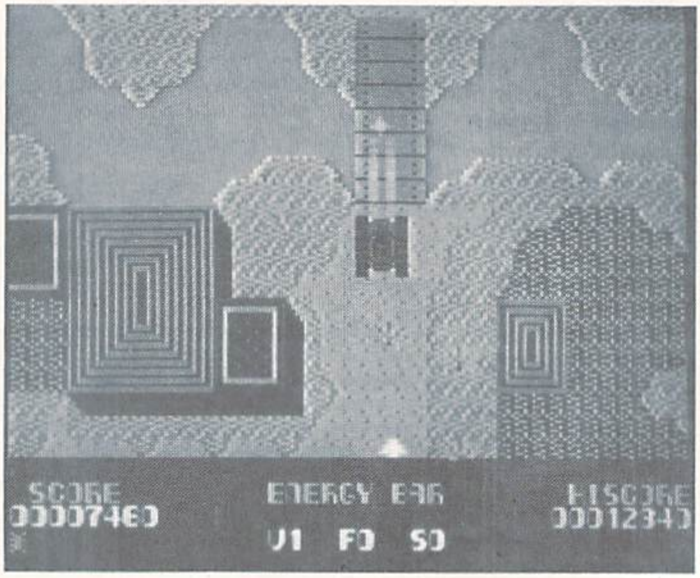
Sex Vixens from Space comprises three C-64 text games for adults on a single disk. In the title game, the player must locate a planet of beautiful, sex-starved female clones. *Bite of the Sorority Vampires* sends a reporter for the college newspaper out to investigate the occult activities of the girls of Eta Beta Pi. *Hatchet Honeymoon* requires the player to outwit a madman, soothe a lusty maid, and consummate the wedding in order to inherit a fortune. You must be 18 years of age or older to order the disk, which sells for $29.95 (shipping free).

Free Spirit Software, Inc., 312-352-7323 (see address list, page 14).

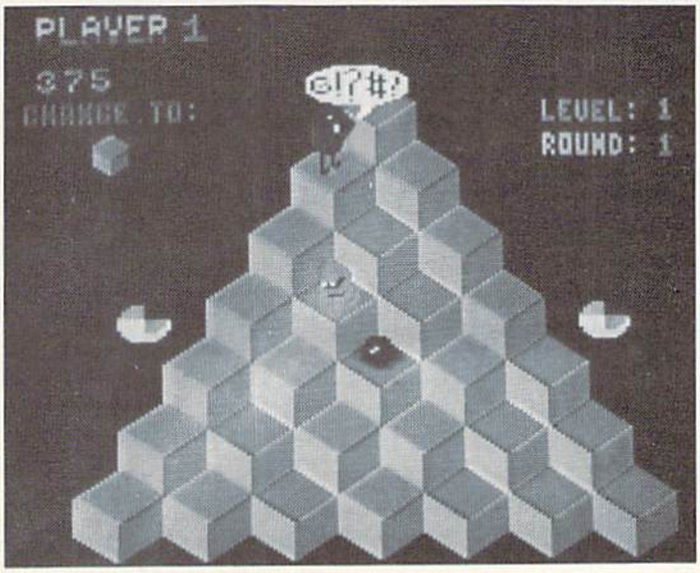
NEW GAMES

Two additions to Data East's Moving Target line of coin-op translations for the C-64, each $9.95:

TNK III puts the player in command of an experimental supertank, with advanced weaponry that must be used to



As commander of the experimental **TNK III,** *the player becomes a one tank army facing all manner of enemy firepower, from anti-tank guns to submarines.* READER SERVICE NO. 126



The arcade classic* Q*Bert *has been released previously for the C-64—but at $9.95, who could resist the already irresistible hopping hoozit? READER SERVICE NO. 127

defeat enemy tanks, submarines, antitank gun emplacements, and guerilla troops.

As is well known to most people who've seen the inside of a videogame arcade, *Q*Bert* springs up and down the steps of a pyramid, changing the colors of each step as he lands. The object is to color in nine pyramids, avoiding the numerous creatures blocking the way.

Data East, 408-286-7074 (see address list, page 14).

For the C-64 from Scorpion, each $29.95:

Alien Destruction Set consists of four arcade adventures in space, requiring skills ranging from quick puzzle-solving logic to fast space-combat reflexes.

In one of the two arcade adventures that comprise *Mandroid*, the half-man, half-machine title character must infiltrate an enemy complex to save seven astronauts and restore their radio contact with earth. In the other, the player must stop an evil scientist from using the designs for Mandroid to create an army of deadly soldiers.

Scorpion, 201-663-0202 (see address list, page 14).

Jinxter, a humorous graphic adventure, is set in a land protected by a magical charm bracelet that limits the power of a secret society of Green Witches. But the charms have been dispersed and the witches are running wild. You must find and link the seven charms. $34.95 for the C-64; $39.95 for the Amiga.

Rainbird Software/Activision, 415-960-0518 (see address list, page 14).

New for the C-64 by Free Spirit:

Based on the popular big wheel car and truck competitions, *Monster Power* ($14.95) lets one to four players compete in Tractor Pulls, Mud Bogs, and Monster Trucks.

In *Strategic Playground Football* ($9.95), one or two players can compete using playground rules. Though offensive and defensive strategies are selected via keyboard input, they can go awry as a dog runs across the field or players leave the game to catch the ice cream truck.

A strategy simulation of the manufacturing of robots in a futuristic society, *Univeral Robots* ($9.95) requires you as plant manager to produce and sell new robot models at a profit.

Eye of the Inca ($19.95) contains four text adventures: the title game, in which you scour an ancient temple for a gigantic diamond; *Shipwrecked*, requiring you to survive as a castaway on a south seas island; *Son of Ali Baba*, wherein you battle an evil magician and his army of monsters in old Baghdad; and *Perils of Darkest Africa*, all about a search for jewels from King Solomon's mines.

Revenge of the Moon Goddess ($19.95) also contains four text adventures: the title game, sending you into the heart of the South American jungles in quest of the lost City of the Sun;

Frankenstein's Legacy, containing cadavers, deserted old mansions, cemeteries, abandoned windmills, werewolves, and of course The Creature; *Night of the Walking Dead,* sending you in search of the grave of Aunt Bedilla in a zombie-infested cemetery; and *Sea Phantom,* in which the stormy Atlantic provides the backdrop for a tale of ghost ships, mysterious sea caves, haunted mansions, and a restless spirit.

Three Hours to Live ($9.95), a text adventure on a disk all by itself, awakens the player in an immense alien maze. The air has been poisoned, leaving the player three hours to escape.

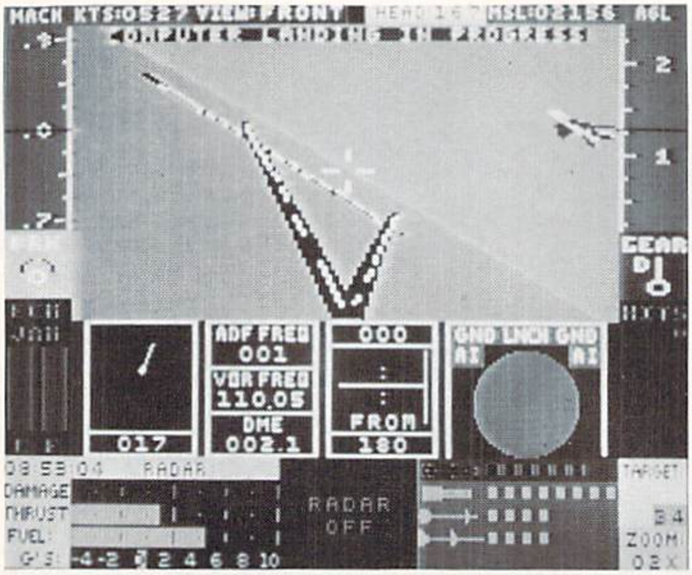
Free Spirit Software, Inc., 312-352-7323 (see address list, page 14).

Stealth Mission ($49.95) provides C-64 flight simulator fans with a choice of three jets: an F-19 Stealth fighter, the experimental forward-swept wing X-29, and a Navy F-14 Tomcat. There are also eight missions and ten skill levels to choose from. Multiple external viewpoints are available. A targeting computer helps you detect, track, and lock onto enemy targets.

SubLOGIC Corporation, 217-359-8482 (see address list, page 14).

For the C-64 from SSI, $39.95 each:

The Eternal Dagger, sequel to *Wizard's Crown,* sends you and your companions in search of the Demon Portal and the enchanted weapon that can seal it. The game's combat system allows you to let the computer resolve each battle quickly, or personally direct the action. Characters from *Wizard's Crown* can be transferred to the sequel.



Stealth Mission's *3D animation techniques provide faster frame rates for cockpit views. Multiple external viewpoints are available.* *READER SERVICE NO. 133*

Shiloh: Grant's Trial in the West consists of Introductory, Intermediate, and Advanced Games that recreate the pivotal Civil War battle in 15 turns. All brigades that were present are included, as well as the Federal gunboats Lexington and Tyler. The 30x30 square grid battlefield accurately reflects the terrain around Shiloh.

Strategic Simulations, Inc., 415-964-1353 (see address list, page 14).

Plasmatron ($14.95) assigns the player the mission of ridding a hostile empire colony of the countless aliens that inhabit it. On the way, he flies his Plasmatron craft through futuristic cities and meteor fields.

Accolade, 408-446-5757 (see address list, page 14).

Bull Run to Chancellorsville, Volume I in the Strategic Studies Group/Electronic Arts series, *Decisive Battles of the American Civil War,* presents six simulations: the First and Second Bull Run, Shiloh, Antietam, Fredericksburg, and Chancellorsville. The series utilizes a system that recreates the military structure through your eyes as Army Commander of Corps, Divisions, and Brigades. Each command level has its own Artificial Intelligence structures, providing you with real life subordinates. You can elect to use the command and sighting rules, or personally command all units.

Electronic Arts, 415-571-7171 (see address list, page 14).

Play Bridge with Truscott ($29.95) is adapted from Dorothy Hayden Truscott's book which reveals her secrets of winning declarer play. The C-64 user is given from one to five chances to find the correct play in each of 158 deals illustrating suit play, notrump play, safety plays, card reading, end plays, squeezes, and deceptions.

Great Game Products, 800-GAMES-4-U (see address list, page 14).

GAMING CLUES

Quest for Clues ($24.99) provides solutions to 50 of the most popular text adventures and fantasy role-playing games. Included are over 80 detailed maps for players lost in fantasy worlds. All the major clues are encoded to prevent readers from accidentally reading more clues than they need. An overview of each game is also provided, to help readers decide before buying a game if it's one they'll enjoy.

Origin Systems, Inc., 603-644-3360 (see address list, page 14).

The Commodore Hint Disk ($10.50) includes pointers for playing *The Pawn, Guild of Thieves, Leather Goddesses, Bard's Tale* and *Might & Magic.*

Elkon Enterprises (see address list, page 14).

THAT'S THE SPIRIT

We're pleased to announce a victory for the good guys in the dispute between Free Spirit Software and Keypunch Software (see March '88 *Scuttlebutt).* Under the terms of the settlement, Keypunch acknowledges that



80 pages of software, MIDI items.
READER SERVICE NO. 134

Free Spirit owns the copyright to the eight games involved, and will pay substantial damages to Free Spirit for selling them without their consent. But, settlements being what they are, Free Spirit will license Keypunch to market these games in exchange for royalties. Contact Free Spirit for additional information.

Free Spirit Software Inc., 312-352-7323 (see address list, page 14).

MUSIC CATALOG

The third edition of The Musician's Music Software Catalog contains over 80 pages of software and MIDI interfaces from various manufacturers for various computer systems, among them the C-64. The detailed product descriptions provided for all items, along with the screen images and transcription examples for many, can help the consumer choose the best product based on price and capability. Free telephone consultation is offered. A two-year catalog subscription costs $3.00 (refundable with your first order).

Digital Arts & Technologies, 800-332-2251 or 203-874-9080 (see address list, page 14).

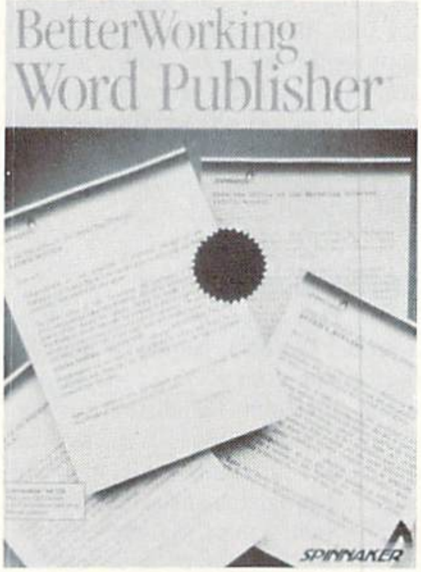
C-128 CP/M DISKS

To paraphrase another famous saying: no one talks about CP/M on the C-128, or does anything about it either. At least, it seems that way in between arrival of programs that support the 128's neglected third mode. The latest, *Jugg'ler 128* ($17.95) provides read, write, and formatting support for more than 130 types of MFM CP/M disks on the C-128 in CP/M mode with a 1570, 1571, or 1581 disk drive. It is compatible with all current versions of C-128 CP/M. Special features include three user-definable default disk types, access to most menu options directly from the CP/M command line, and the ability to analyze an unknown CP/M disk type and report possible matches from an internal database of disk types.

Transactor Publishing Inc., 416-764-5273 (see address list, page 14).

GEOS WP

Spinnaker asserts that *Betterworking Word Publisher* ($39.95), their GEOS word processor for the 64, enters text five times faster than similar products. The program also chains files for unlimited document size, and includes a 10,000 word spell checker. Up to nine fonts and six typestyles can be used per page, and compatibility with *geoPaint* and *geoFont* allows additional graphics and fonts to be imported.



A fast entry GEOS word processor.
READER SERVICE NO. 135

Spinnaker Software, 617-494-1200 (see address list, page 14).

BUY TWO, GET ONE FREE

From April 1 through June 30, consumers who buy two Electronic Arts

programs will be able to select a third for free. You can obtain a coupon at your local retail outlet, which must be sent in by July 15 along with proof of purchase and $3.00 for shipping and handling.

Electronic Arts, 415-571-7171 (see address list, this page).

CHILD SOFTWARE

The *1988 Survey of Early Childhood Software* ($20.00), published annually by the nonprofit High/Scope Educational Research Foundation, is designed to help parents and teachers find the correct programs for 3- to 6-year olds. 286 programs for various computers, including the C-64, are rated for ease of use, educational value, instructional design, skills required, and amount of adult supervision needed.

High/Scope Educational Research Foundation, 313-485-2000 (see address list, this page).

128 REPAIR GUIDE

Commodore 128 Troubleshooting & Repair offers hobbyists, service technicians, and operators at all levels the information necessary to maintain and repair equipment. Covered are diagnostic techniques, operation, hardware, preventive maintenance, required tools, data sheets, and assembly and disassembly instructions. The inner workings of the 128 are explained, along with methods for avoiding problems through routine care and maintenance. 160 pages; $19.95

Howard W. Sams & Company, 317-298-5722 (see address list, this page).

TAKE OFF THE EDGE

Easystrip ($14.95) removes the perforated edges from as many as 30 sheets of fanfold paper at once. The device, resembling a hinged ruler, is flipped open, and the sprocket holes are dropped over the pegs. It is then closed and the sheets torn free, leaving the edges behind. Easystrip is made of plastic, with foam padding on the bottom to prevent slipping.

Oremco, Inc., 212-246-1429 (see address list, this page).

FEDERAL EXPRESS

The latest in the Blue Lion line of travelogues, *Ticket to Washington, DC* ($34.95) takes the C-64 user to 45 different locations in the US capital, from museums and Capitol Hill to Georgetown's cobblestones and Mount Vernon. You accomplish this sightseeing on your way to discovering the identity of a famous American from the past or present, following clues that lead you all over town. Seven 3-D maps and onscreen information aid your search. 14 different mysteries are included.

Blue Lion Software, 617-876-2500 (see address list, this page).



Sure, you could remove the perforated edges of your fanfold paper by hand. But Easystrip can be viewed as a computer age status symbol.
READER SERVICE NO. 149

Companies Mentioned in *Scuttlebutt*

Contact manufacturers for more information.

Abacus
5370 52nd Street SE
Grand Rapids, MI 49508
Phone: 616-698-0330

Accolade
20813 Stevens Creek Blvd.
Cupertino, CA 95014
Phone: 408-446-5757

Blue Lion Software
90 Sherman Street
Cambridge, MA 02140
Phone: 617-876-2500

CompuServe
5000 Arlington Centre Blvd.
P.O. Box 20212
Columbus, OH 43220
Phone: 614-457-8600

Data East USA, Inc.
470 Needles Drive
San Jose, CA 95112
Phone: 408-286-7074

Didatech Software Ltd.
3812 William Street
Burnaby, B.C.
Canada V5C 3H9
Phone: 604-299-4435

Digital Arts & Technologies
P.O. Box 11
Milford, CT 06460
Phone: 203-874-9080

Electronic Arts
1820 Gateway Drive
San Mateo, CA 94404
Phone: 415-571-7171

Elkon Enterprises
2914 Pennsylvania
Wichita Falls, TX

Free Spirit Software
905 W. Hillgrove, Suite 6
LaGrange, IL 60525
Phone: 312-352-7323

Great Game Products
8804 Chalon Drive
Bethesda, MD 20817
Phone: 800-GAMES-4-U

High/Scope
600 North River Street
Ypsilanti, MI 48198-2898
Phone: 313-485-2000

Howard W. Sams & Co.
4300 West 62nd Street
Indianapolis, IN 46268
Phone: 317-298-5400

Inkwell Systems
5710 Ruffin Road
San Diego, CA 92123-1013
Phone: 619-268-8792

Mindscape Inc.
3444 Dundee Road
Northbrook, IL 60062
Phone: 312-480-7667

Oremco, Inc.
261 Madison Avenue
New York, NY 10016
Phone: 212-246-1429

Origin Systems, Inc.
136 Harvey Rd.—Bldg. B
Londonderry, NH 03053
Phone: 603-644-3360

Rainbird Software
c/o **Activision**
2350 Bayshore Parkway
Mountain View, CA 94043
Phone: 415-960-0410

S.O.G.W.A.P. Software
115 Bellmont Road
Decatur, IN 46733
Phone: 219-724-3900

Spinnaker Software
One Kendall Square
Cambridge, MA 02139
Phone: 617-494-1200

Strategic Simulations
1046 N. Rengstorff Ave.
Mountain View, CA 94043
Phone: 415-964-1353

SubLOGIC Corporation
713 Edgebrook Drive
Champaign, IL 61820
Phone: 217-359-8482

Transactor Publishing
85 W. Wilmot St., Unit 10
Richmond Hill, ONT
Canada L4B 1K7



For the C-128

By Cleveland M. Blakemore

Rashgar is the immortal champion of the X dimension and keeper of the gate through reality.

Nobody knows how long Rashgar has been there, guarding the locus of the X dimension against the evil Crystallions, the race who would break through the barrier and destroy the universe. He cannot be annihilated, only temporarily disembodied.

Rashgar is a fast-paced arcade game for the C-128 that requires a joystick in Port 2.

The object is to guard the center of the screen (and yourself) from the Crystallions, spinning, constantly pulsating crystals that float dreamily through the domain of Rashgar. To destroy one of these creatures, center the target cross hair on it using the joystick and press the fire button. Rashgar's eyes light up with fury, and a lance of energy leaps from his mouth and blasts the menace into subatomic vapor. Both the cross hair movement and the laser beam wrap around the screen.

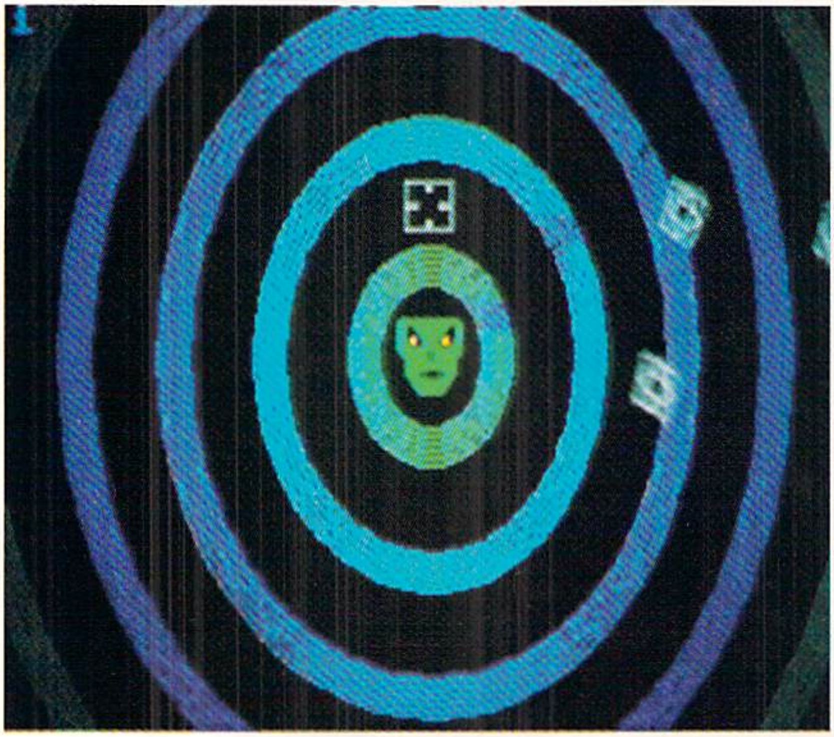
The cross hairs represent Rashgar's aiming point for the laser weapon in his mouth, so don't touch him with it, or he'll fire at himself!

At first, the Crystallions will be slow and easily hit, but with time they will increase in speed and frenzy until they enter Rashgar's safety zone and destroy him. He has three lives before the game ends.

The current score is displayed at the top of the screen over Rashgar's head. The level is in the left corner, and the number of lives remaining for Rashgar is displayed in the upper right. My high score to date is 27,200 points, well past the eighth level of difficulty.

Although *Rashgar* is only 16 blocks long, and the shortest game I have ever written for *Ahoy!*, I think you'll be amazed at how much action and graphic excitement is squeezed in. I used BASIC 7.0 commands to create the sprite shapes for the Crystallion creatures, which saved an enormous amount of code that would normally be wasted on DATA statements. What data was required was stored as hexadecimal numbers, thus taking up half as much space as decimal data.

The short interrupt routine in the game animates the Crystallions by flipping through their sprite patterns, and also makes Rashgar curse the evil creatures.



PHOTOS: MICHAEL R. DAVILA

You've probably heard that the C-128's DRAW command is so swift it could be used to create a laser beam for an arcade game, and *Rashgar* proves it. I think you'll agree it's one of the quickest line commands anybody ever saw in a BASIC interpreter.

The giant letters for the GAME OVER prompt were created using the CHAR commands with SPRSAV, and then expanding the sprites to twice normal size. This also saved a tremendous amount of code.

You'll notice that strategy plays a small part here. Concentrate on the creatures which have been onscreen the longest, or are getting too close, and zap them first. Work at surviving before scoring, and you'll make it to the higher levels.

If you've never typed in a program for your C-128 from *Ahoy!* before, try *Rashgar*. It's a heckuva lot in a small package. ☐

SEE PROGRAM LISTING ON PAGE 70

TEVEX Computer Software 1-800-554-1162

SSI

SSI	LIST PRICE	OUR PRICE
B-24	$35	$24
Battlecruiser	$60	$41
Gettysburg	$60	$41
Kampfgruppe	$60	$41
Mech Brigade	$60	$41
Panzer Strike	$45	$31
Phantasie I or II, or III	$40	$28
President Elect '88	$25	$18
Questron II	$40	$28
Realms of Darkness	$40	$28
Rebel Charge	$50	$34
Rings of Zilfin	$40	$28
Roadwar 2000	$40	$28
Roadwar Europa	$40	$28
Shard of Spring	$40	$28
Shiloh	$40	$28
Sons of Liberty	$35	$24
Wargame Constr.	$30	$21
War South Pacific	$60	$41
Wizard's Crown I or II	$40	$28

MICROPROSE

MICROPROSE	LIST PRICE	OUR PRICE
Airborne Ranger	$35	$24
F-15	$35	$24
Gunship	$35	$24
Pirates	$40	$28
Silent Service	$35	$24
Stealth Fighter	$40	$28

EPYX

EPYX	LIST PRICE	OUR PRICE
California Games	$40	$28
Champ. Wrestling	$40	$28
Destroyer	$40	$28
Movie Monster	$20	$15
Spy vs Spy III	$25	$18
Street Sports (each)	$40	$28
Sub Battle	$40	$28
The Games - Winter ed.	$40	$28
World Games	$40	$28
EPYX JOYSTICK	$22	$15

SSG

SSG	LIST PRICE	OUR PRICE
Battlefront	$40	$28
Battles in Normandy	$40	$28
Battles of the Civil War	$40	$28
Carriers at War	$50	$34
Halls of Montezuma	$40	$28
Reach for the Stars	$45	$31
Russia	$40	$28
Run 5 Magazine #8		$4

ACCOLADE

ACCOLADE	LIST PRICE	OUR PRICE
Apollo 18	$30	$21
Card Sharks	$30	$21
4th & Inches	$30	$21
Hardball	$30	$21
Mini Putt	$30	$21
Plasmatron	$15	$12
Power at Sea	$30	$21
Spy vs Spy I & II	$15	$12
Test Drive	$30	$21
The Train	$30	$21

INFOCOM

INFOCOM	LIST PRICE	OUR PRICE
Beyond Zork - 128k	$45	$31
Border Zone	$35	$24
Lurking Horror	$35	$24
Nord & Bert	$35	$24
Sherlock	$35	$24
Stationfall	$35	$24
Invisiclues - call for availability		

ACTIVISION

ACTIVISION	LIST PRICE	OUR PRICE
Aliens	$35	$24
Champ Basketball 2 on 2	$35	$24
Champ Football	$35	$24
Labyrinth	$30	$21
Last Ninja	$35	$24
Maniac Mansion	$35	$24

AMIGA

AMIGA	LIST PRICE	OUR PRICE
Alien Fires	$40	$28
Balance of Power	$50	$34
Breach	$40	$28
Breach Scenario	$25	$18
Defender of the Crown	$50	$34
Ferrari Formula One	$50	$34
Fire Power	$25	$18
Hunt for Red October	$50	$34
Jet Simulator	$50	$34
King of Chicago	$50	$34
Moebius	$60	$41
Return to Atlantis	$50	$34
Roadwar 2000	$40	$28
Silent Service	$40	$28
Sinbad	$50	$34
Starfleet I	$55	$38
Terrorpods	$40	$28
Test Drive	$45	$31
Uninvited	$50	$34
Weaver Baseball	$50	$34
Winter Games	$40	$28

ELEC. ARTS

ELEC. ARTS	LIST PRICE	OUR PRICE
Adventure Constr. Set	$15	$12
Amnesia	$40	$28
Arcticfox	$33	$23
Bard's Tale I or II	$40	$28
Chessmaster 2000	$40	$28
Chuck Yeager's AFT	$35	$24
Dragon's Lair	$25	$18
Earth Orbit Station	$30	$21
Legacy of the Ancients	$30	$21
Lords of Conquest	$15	$12
Marble Madness	$30	$21
Pegasus	$30	$21
Skate or Die	$30	$21
Seven Cities of Gold	$15	$12
Skyfox II	$30	$21
Strike Fleet	$30	$21

ORIGIN SYS.

ORIGIN SYS.	LIST PRICE	OUR PRICE
Auto Duel	$50	$34
Ogre	$30	$21
Ultima I	$40	$28
Ultima III	$50	$34
Ultima IV or Moebius	$60	$41

ETC.

ETC.	LIST PRICE	OUR PRICE
AR - City or Dungeon	$30	$21
Blue Powder, Grey Smoke	$50	$34
Borodino	$60	$42
Choplifter-Mid. Magic	$15	$12
Dark Horn	$35	$24
Defender of the Crown	$35	$24
Echelon	$45	$31
Full Count Baseball	$40	$30
Gauntlet	$35	$24
Guderian	$30	$21
High Seas	$50	$34
Ikari Warrior	$35	$24
Kid Niki	$30	$21
Micro League Baseball	$40	$28
Might & Magic	$40	$28
NBA	$40	$28
Paperboy	$35	$24
Star Fleet I	$40	$28
Stealth Mission	$50	$34
Up Periscope	$30	$21
Where in U.S.Carmen?	$35	$24
Wizardry I	$40	$28
World Class Leader Board	$40	$28
Wooden Ships & Iron Men	$35	$24
Wrath of Denethenor	$20	$15
WWF Wrestling	$30	$21



Same Day Shipping

Just call before 3:30 and we'll ship your order *today* by UPS. Your package is only days away with Tevex.

2 for 1 SALE! Buy 2 get 1 FREE for selected Electronic Arts software. Call for details.

CALL TOLL-FREE 1-800-554-1162

FREE CATALOG WITH YOUR FIRST ORDER

Open 9-8 Mon. - Fri. 10-5 Sat.
Retail Store open same hours.

TEVEX

4205 First Ave, Suite 100
Tucker (Atlanta), GA 30084
404-934-5059

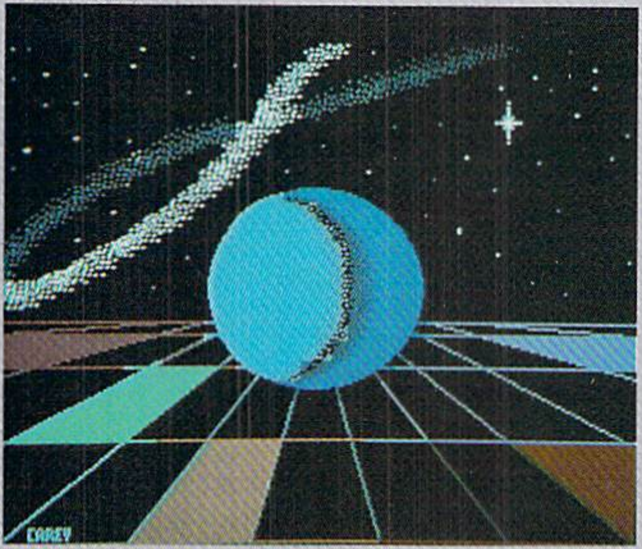
Georgia residents call 404-934-5059. When ordering by mail send money order. Include phone number. SHIPPING: Add $3.00 for UPS shipping and handling charge. Georgia residents add 4% sales tax. Shipping for Canadian orders is 5% of order, with a $4.00 minimum. U.S. Mail, APO & FPO orders add 5% of order, with a $4.00 minimum. Shipping for all other foreign orders is 15% of order, with a $10.00 minimum. All sales are final.

New Titles are underlined

VISA MasterCard

Reader Service No. 148

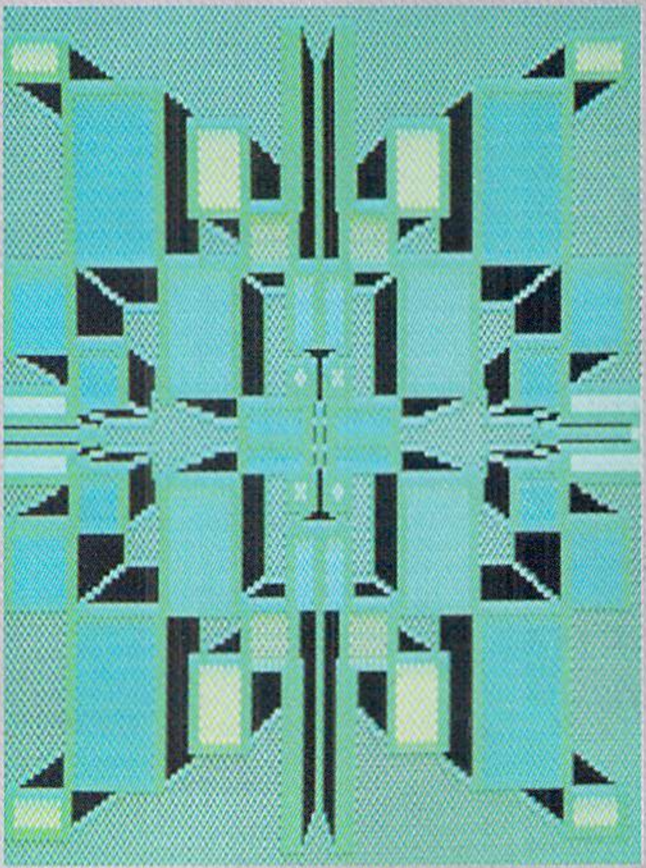
ART GALLERY

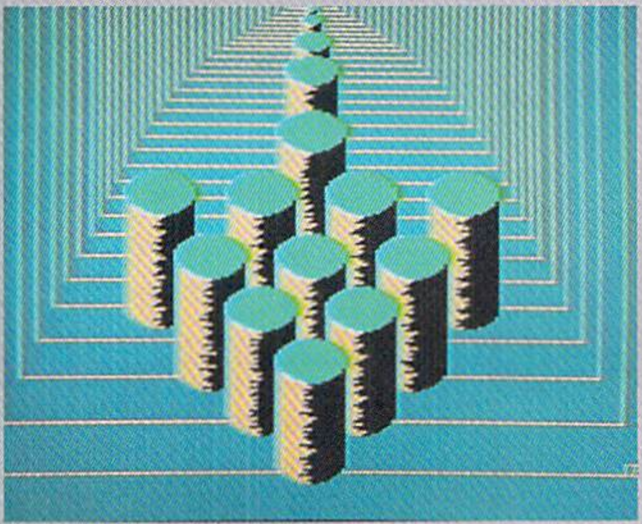


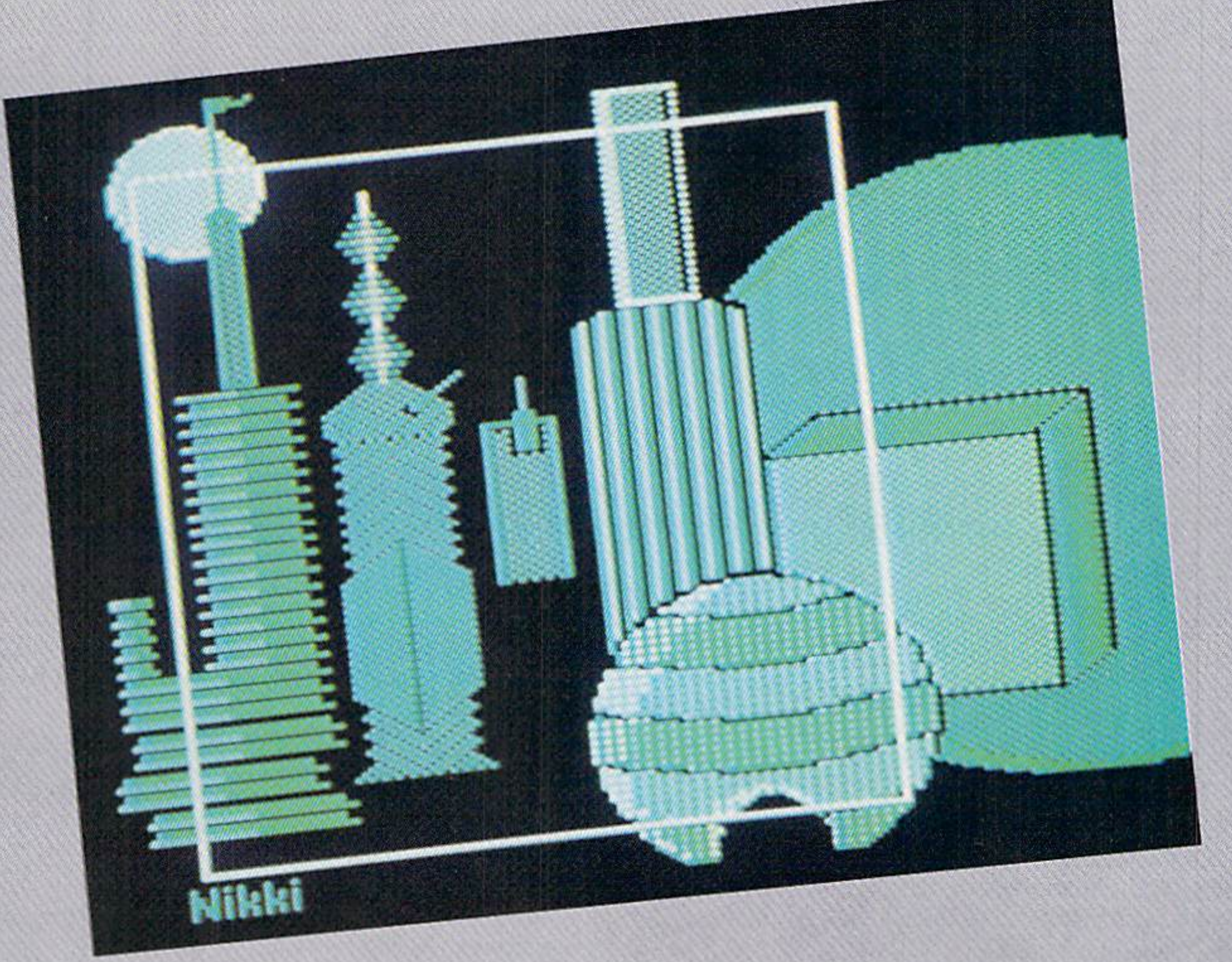
The *Ahoy! Art Gallery* offers the opportunity for fame and fortune to aspiring Commodore artists. Send your work on disk to Morton Kevelson, P.O. Box 290260, Homecrest Station, Brooklyn, NY 11229-0005. Indicate the drawing package or file format of the images. Graphics produced on the Amiga are eligible for inclusion in *Ahoy!'s AmigaUser*: C-64, C-128, and Plus/4 images are eligible for inclusion in *Ahoy!* If your image is published, you will receive a free one-year subscription. Current subscribers will have their subscription extended by one year.

Note that the *Art Gallery* is not a contest. Published pictures are selected in an arbitrary and capricious fashion by the *Ahoy!* Art Director, based solely on their artistic merit.

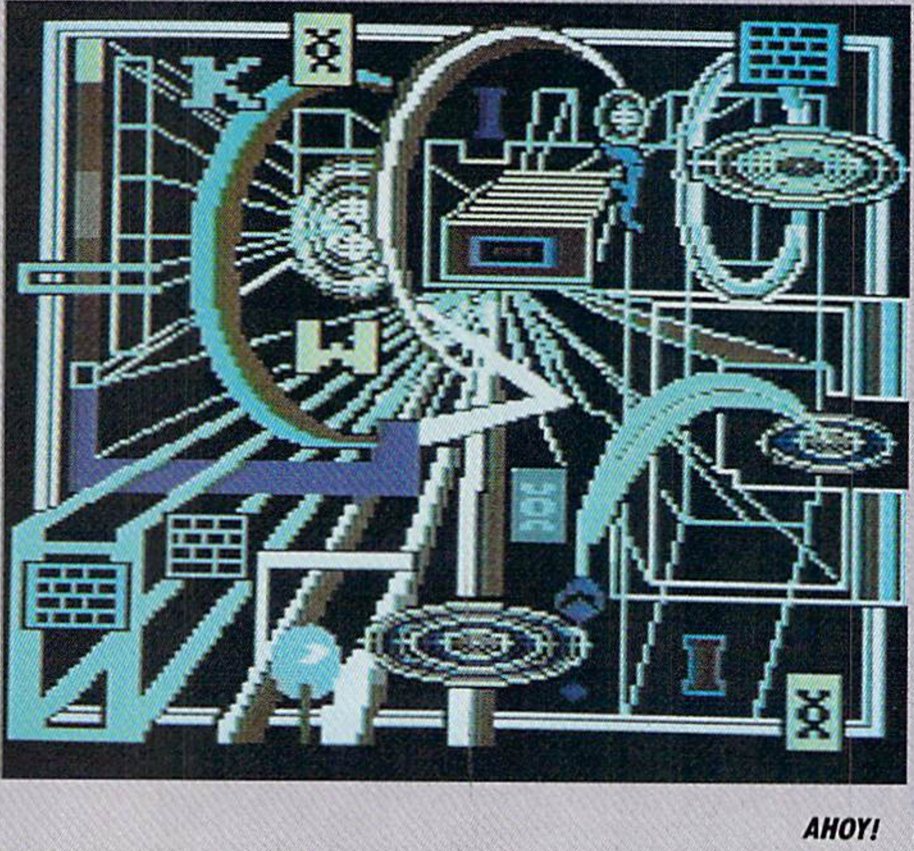


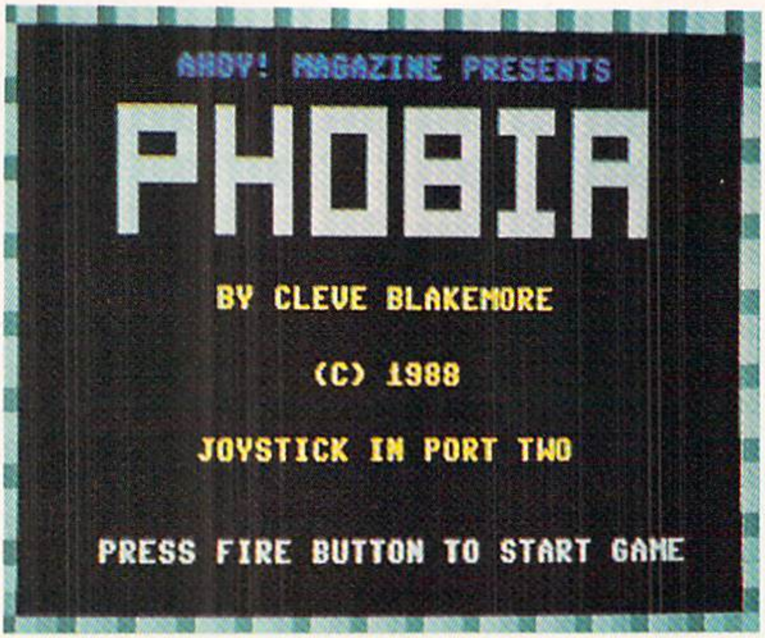






If some of this month's images are a little abstract for your taste, pretend you like them. It's great practice for the next time you find yourself in a modern art museum with a snobbish date. Images were created on *Koala* except where noted. At bottom left of the facing page is *Design #7,* by Barbara Tiess (Middletown, NY). Right half of facing page, top to bottom: catch the *Ball* thrown together by Cary Hildebrand (Morden, Manitoba) on *DOODLE!*; enter the *Doorway* laid open by A. Black (Santa Rosa, CA); and go down the *Tubes* formed with *Sorcerer's Apprentice* by Doug Barnett (Mountain View, CA). We regret the lack of a yellow brick road to guide you to the top of this page, and *Oz* by graphics wiz Nikki Lewis (Des Plaines, IL). At right is *Abstract* by Gerald J. Abear (Las Cruces, NM), drawn on CADPIC.





For the C-64

No one forced you to enter the Reality Institute. You volunteered of your own free will. You were curious to run what you had heard called the "Ultimate Obstacle Course," a nigh-lethal challenge of both mind and body, an elemental test of the lifeforce itself.

Even after getting over the pits of deadly chemicals and past the laser fire from the security system, you will have to answer questions requiring such tremendous self-introspection and judgment that they dwarf any test you have ever taken in your life. Worst of all, you must answer these questions before the moving wall behind you crushes you into pulp.

Anybody can go in. But can anybody come out?

I wrote *Phobia* for five specific reasons:

1. To do another offbeat game that would surpass *Teleporter* (Nov. '86) in originality. I wanted to do a game that would test left/right brain coordination the way *Teleporter* tested intuition.

2. To satisfy our friends over at *INFO* Magazine who have been clamoring for more good all-BASIC games from the computer magazines, rather than just machine language listings which don't teach.

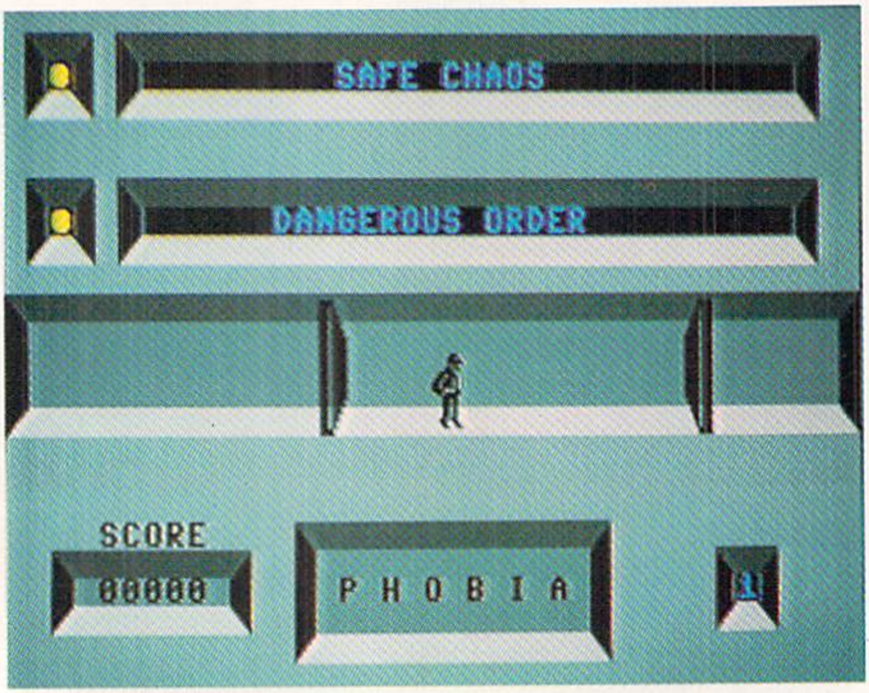
3. To finally demonstrate the nature of left-right brain interaction, and create a game that would test not just a person's knowledge but his actual ability to THINK (and under pressure at that).

4. I had just finished *Masters of Darkness*, a new short story collection from Tor Books, and was so enthralled with Fritz Leiber's *Black Corridor* that I just had to write a computer game with this wonderful short as inspiration.

5. I was curious to see if I could do a really worthwhile arcade game in BASIC 2.0. I have never done an action game completely without machine language before for *Ahoy!*, and I had a handful of really keen tricks for speed running around inside my head. After seeing *Hanger 14* in the Jan. '87 issue, I suspected you could do quite a bit of animation without a whit of object code, if you really used your noggin. Congratulations to author Justin Luton for his terrific demo of a BASIC arcade game without ML.

The game uses Port 2 for joystick control. The left and right directions move your man back and forth, the fire button allows him to jump, and the up and down motion of the joystick permits you to choose between the two categories of ideas displayed on the giant computer screens overhead.

Your simple objective is to survive the Reality Corridor, a narrow tunnel filled with trenches of noxious chemicals and rapid laser fire from ahead. The wall to your rear is constantly sliding forward, threatening to crush you to death against the wall ahead. If you stand in place, the trailing wall will push you into the barrier, or over into one of the pits. You must keep moving!



The laser fire will incinerate you if it hits you. Press the fire button and leap over these photon lances when they approach.

Before you reach the door at the end of the corridor, you must choose between the two concepts displayed above. Moving the joystick up or down will illuminate the lights on the panels beside the screens, with the

PHOTOS: MICHAEL R. DAVILA

Ahoy! DISK

Why type in the listings in this month's *Ahoy!* when we've done it for you? All the programs in this issue are available on the current *Ahoy! Disk* for $8.95. Isn't your time worth more than that?

Use the coupon at the bottom of this page to order disks for individual months, a disk subscription, or the special anthology disks described below. (You can also subscribe to the *Ahoy! Disk Magazine*— disk and magazine packaged together at a special reduced rate. See the card bound between pages 50 and 51.)

BEST OF '84
Address Book (3/84)
Air Assault (5/84)
Alpiner (6/84)
Checklist (7/84)
Construction Co. (12/84)
DOS (9/84)
Emerald Elephant (10/84)
I Got Biorhythm (5/84)
Lawn Job (10/84)
Lunar Lander (4/84)
Math Master (5/84)
Micro Minder (8/84)
Multi Draw 64 (1/84)
Name That Star (4/84)
Post Time (6/84)
Salvage Diver (9/84)
Space Lanes (VIC) (3/84)
Space Lanes (64) (3/84)
Space Patrol (12/84)
VIC 40 (10/84)

BEST OF '86
Disk Cataloger (8/86)
Dogcatcher (9/86)
Fidgits' Music School (7/86)
Flap! (8/86)
Guardian (11/86)
Head to Head (2/86)
Highlight (8/86)
JCALC (4/86)
Meteor Run (8/86)
Mine Canyon (9/86)
Mountaineer Mack (10/86)
Star Strike (6/86)
Step On It (8/86)
Syntax Patrol (9/86)
Teleporter (11/86)
Tenpins (11/86)
1541 Alignment Check (2/86)

BEST UTILITIES II
Alarm Clock (1/86)
BASIC Relocator (1/86)
Bigprint (5/86)
Chrono-Wedge (4/86)
Disk Checkup (10/86)
Failsafe (5/86)
Long Lines (11/86)
Programmer's Aid (3/86)
Quick Change (10/86)
Reversed Remarks (7/86)
Screen Sleuth (7/86)
Screen Window (4/86)
Speech64 (9/86)
The Editor (12/86)
Trackdown (6/86)
Very-Able (3/86)
128 to 64 Autoboot (10/86)

BEST GAMES II
Actionauts (4/86)
Air Rescue (4/86)
Brickbusters (3/86)
Crabfight (2/86)
English Darts (5/86)
Got a Match? (6/86)
Haunted Castle (1/86)
Head On (7/86)
Knockout (1/86)
Micro City (12/86)
Minotaur Maze (12/86)
Penguins (10/86)
Ski Folly (5/86)
Star Search (5/86)
Swoop (4/86)
The Last Ninja (9/86)

BEST OF '85
Ahoy! Dock (9/85)
Auto Line Numbers (3/85)
Directory Manipulator (12/85)
File Lock (9/85)
Fontasia (2/85)
Futurewar! (2/85)
Gameloader (12/85)
Home Budget (3/85)
Jewel Quest (12/85)
Lightning Loader (11/85)
Midprint (1/85)
Music Tutor (1/85)
Salmon Run (3/85)
Screen Dump (8/85)
Telelink (2/85)
Tile Time (5/85)

BEST UTILITIES
Auto Append (7/85)
BASIC Trace (4/85)
Base Conversions (9/84)
Booter (4/85)
Commodore Font Editor (5/85)
DOS (9/84)
Fastnew (9/85)
Formatter (8/85)
Lucky Lottery (7/85)
Maestro (10/85)
Micro Minder (8/84)
Printat (7/85)
Screen Bit Dump (6/85)
Screen Magic (6/85)
Start & End File Address (9/85)
Two Column Directory (5/85)
Windows (8/85)
1541 Cleaning Utility (10/85)

BEST GAMES
Alpiner (6/84)
Construction Co. (12/84)
Disintegrator (8/85)
Dragon Type (9/85)
Duck Shoot (6/85)
Emerald Elephant (10/84)
Gators N Snakes (8/85)
Gravinauts (10/85)
Lawn Job (10/84)
Mystery at Mycroft Mews (10/85)
Name That Star (4/84)
Post Time (6/84)
Salvage Diver (9/84)
Script Analysis (7/85)
Space Patrol (12/84)
Tile Time (5/85)
Wizard of Im (7/85)

BEST C-128
Ahoy!Term (3/86)
Dark Fortress (1/87)
Dual Dump (9/86)
Extended Background Mode (4/87)
Infoflow (4/87)
List Formatter (4/87)
More BASIC 128 (2/87)
Nebergall Run (7/86)
Planet Duel (4/87)
Queen's and Bishop's Tours (6/86)
Rescue 128 (3/87)
The Artist (12/86)
Variable Manager (9/86)
Wordcount (7/86)
128 to 64 Autoboot (10/86)

We've collected Ahoy!'s best programs onto the Anthology Disks described here. Please note that many of these programs will be unusable without the documentation printed in the issues of Ahoy! listed in parentheses.

SINGLE ISSUE AND SUBSCRIPTION PRICES

(Postage and Handling Included)	U.S.A.	CANADA	ELSEWHERE
❑ June '88 *Ahoy!* Disk	$ 8.95	$ 12.95	$ 13.95
❑ 12-Month Disk Subscription	$ 79.95	$ 99.95	$124.95
❑ 24-Month Disk Subscription	$149.95	$179.95	–

BACK ISSUES $8.95

In Canada add $2.00 per issue; outside US and Canada add $4.00 per issue.

❑ ____	$8.95	❑ ____	$8.95
❑ ____	$8.95	❑ ____	$8.95
❑ ____	$8.95	❑ ____	$8.95
❑ ____	$8.95	❑ ____	$8.95

ANTHOLOGY DISKS $10.95 EACH

❑ Best of '84 ❑ Best Utilities
❑ Best of '85 ❑ Best Utilities II
❑ Best of '86 ❑ Best Games
❑ Best of C-128 ❑ Best Games II

Send coupon or facsimile to: *Ahoy!* Program Disk
Ion International Inc.
45 West 34th Street, Suite 500
New York, NY 10001

MC and VISA Accepted Call 718-979-9330

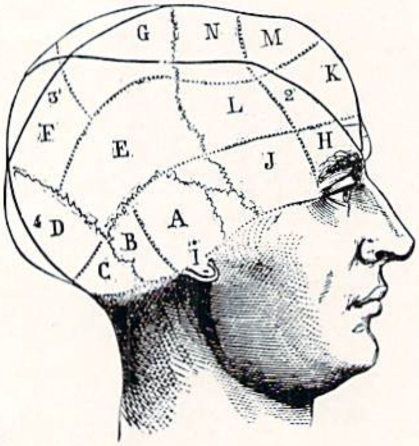
NAME ____________________

ADDRESS ____________________

CITY ____________ STATE ________ ZIP ________

Make check or money order payable to **Double L Software**. N.Y. State Residents must add 8¼% sales tax.

positive choice in green and the negative choice in red. Until you make a choice, the wall in front of you will not budge, and you can't go through it. If you make an incorrect choice, contact with the barrier in front of you will activate a current in the floor beneath and electrocute you. But if you choose wisely, the wall will hum briefly and grind forward away from you into the next room, permitting you to exit the corridor safely and enter the next one.



There are nine corridor sections to traverse. As of this writing, I have made it past the ninth level and graduated from the Reality Institute only once. (And I know all the answers to the questions, too!) The ninth level requires split second timing and instantaneous reaction to survive.

I would like to go into detail on the questions, but it would spoil everything. Instead, I'll only describe them as similar to Zen koans, questions put to eastern students of Taoism in order to test their true capacity to think and comprehend, and to draw their minds into the here and now, rather than drifting off in waking dreams. You must choose the option from which you have *less* to fear. In other words, if you are asked to choose between Fire and Water, you would choose Water, since you might survive being submerged, but you would not survive burning. All the questions in the game concern your choice of survival option, and you must make your decision accordingly. The game gives a good indication of how close you are in touch with reality. If you were asked to choose between Terror and Drowning, you would choose Terror, correct? Fear is a mere emotion, but drowning is a serious proposition indeed.

This sort of understanding requires you to use both the intuitive and analytical portions of the human mind simultaneously, and is a far cry from standard tests of Western intelligence, like "What is the capital of Nevada?" Such questions challenge only one half of the brain and leave the other untouched.

Anybody who wins this game on the first shot, without having seen the questions, would have to be either a kung fu master or the Buddha himself. Don't feel bad about making a lot of mistakes. It's part of the learning curve.

If you manage to learn the correct answers to all the questions in *Phobia,* you might try a real Zen koan, one of the traditional meditation pieces for eastern students: "Is an idea a concept about the real world, or is the real world a concept concerning an ideal?"

This one has kept people occupied for entire lifetimes. □

SEE PROGRAM LISTING ON PAGE 71

FLOTSAM

Send your comments on any aspect of Commodore computing to *Flotsam*, c/o *Ahoy!*, Ion International Inc., 45 West 34th Street—Suite 500, New York, NY 10001.

As mentioned in this month's *View From The Bridge* (see page 7), avid 8-bit users found the news of our cutback to eight C-64/128 issues a year about as welcome as a tax audit. While we can't redress the problem at this point, we can let our readers have their say. Below are selections from letters.

The reasons you gave for going to the Amiga could have been said for going to the Mac—and the new Atari—and every new generation of computer yet to come.

What you obviously don't know is that 7 million C-64 owners still exist and they are more sophisticated and their interests are growing and changing. And, a huge percentage of Amiga owners retain ownership of their 64's. —(name withheld)
Bandon, OR

"Sophisticated" C-64 owners whose interests are "growing and changing"—what an excellent way to describe the Ahoy! *reading audience. As long as that audience is out there, we'll be there for them.*

As soon as one computer comes along, there is another to take its place...in my opinion a user would be better off getting a clone such as the Blue Chip rather than the unpredictable Commodore Amiga. I'm sure many readers would disagree with me—I'm sure many, many more would agree that the C-64 is here to stay.
—Boyd W. Sparks
Kansas City, MO

While few things in the world are here to stay—especially in the microcomputer world—we're counting on the C-64 and C-128 remaining popular for several years to come, and supporting them to the best of our ability.

Why is it that whenever a company gets a good magazine going they end up screwing it up? Is the greed for money (that) overwhelming...? If you want to run a new magazine for the Amiga, then do it, but leave the regular *Ahoy!* alone. I like *Ahoy!* very much...I kept my subscription to it after I let all the others go. Now it looks as if I may have to let it go also. —James E. Pate
Tonopah, NV

We've never had any objection to making a profit. But if "greed for money" were our sole motivation, we'd continue to push out 12 issues of Ahoy! *a year—along with our 4 issues of* Ahoy!'s AmigaUser*—regardless of the dropoff in quality that would result from that extra workload. We've spent the past four and a half years putting out the very best* Ahoy! *we could. We have no interest in putting out any other kind.*

COMPUTER DIRECT

Will Not Be UNDERSOLD!

AND WE MEAN IT!*

Prices Expire 6-30-88

• Famous IBM® Brand •

Genuine IBM® Printer

8 ½" Letter Size

Super Sale $44.95

List $199

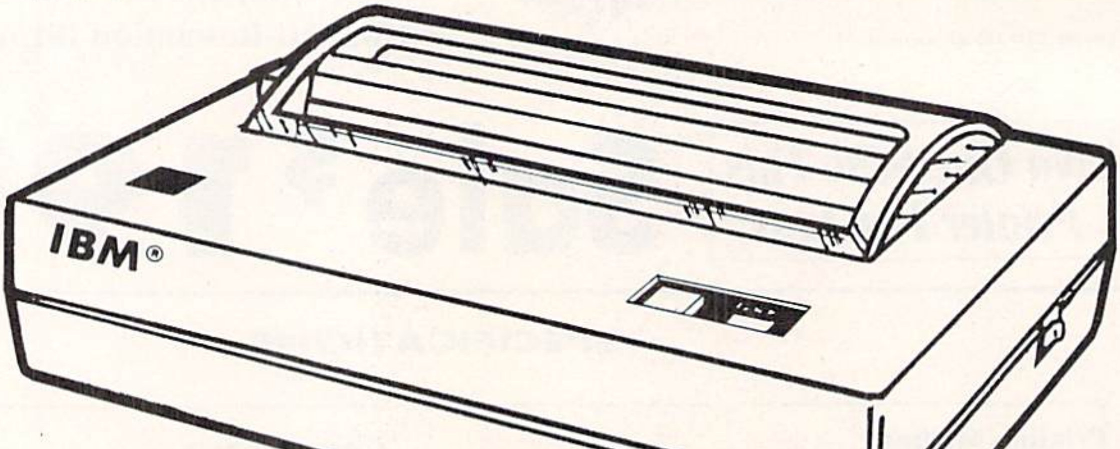
(Add $7.50 shipping.*)

SAVE

Limited Quantities.
1st Come, 1st Serve Basis.

No One Sells This Printer For Less!

Sale



- **80 Column**
- **Enlarged**
- **Underline**
- **Graphics***
- **Unbelievable Low Price!**
- **Advanced Dot Matrix - Heat Transfer**
- **Upper & Lower Case (with True Lower Descenders)**
- **Low Cost Adapters Available For IBM • Apple • Laser • Commodore**

(* Graphics available with Commodore and Apple Interfaces.)

This printer was made by **Canon**® for **IBM**®. It comes ready to hook up to the serial port of the **IBM**®PC jr. Plus with low cost adapter cables you can connect the printer to the **IBM**®PC, XT, AT, & Compatibles, the **Apple**® II,IIe, & IIc, the **Laser**® 128, **Commodore**® 64 ,128, SX-64, Vic 20, & Plus 4, in addition to most MS-DOS Compatibles and more.(See Accessories Page.)

IBM, Apple, Canon, Commodore, Atari & Laser are registered trudemarks of International Business Machines, Apple Computer, Canon Inc, Commodore Business Machines, Atari Inc. & Video Technologies Respectively.

15 Day Home Trial • 90 Day Immediate Replacement Policy

COMPUTER DIRECT
22292 N. Pepper Rd, Barrington, IL. 60010
Call (312) 382-5050 To Order!

* Illinois residents add 6½% sales tax. All orders must be in U.S. Dollars. We ship to all points in the U.S., CANADA, PUERTO RICO, & APO-FPO. Please call for charges outside continental U.S. or C.O.D. **MAIL ORDERS** enclose cashier check, money order, or personal check. Allow 14 days delivery, 2 to 7 days for phone orders and 1 day express mail. Prices and availability subject to change without notice. Shipping and handling charges are not refundable. (Monitors only shipped in continental U.S.) **COMPUTER DIRECT** will match any valid nationally advertised delivered price on the exact same product with a comparable payment method (excluding any applicable sales taxes). A physical copy order of the current valid lower priced ad must be supplied with the order, or within 15 days of date of purchase.

VISA — MASTERCARD — C.O.D.

COMPUTER DIRECT
Will Not Be UNDERSOLD!
AND WE MEAN IT!*

Prices Expire 6-30-88

Super Fast* 15MHz IBM® XT Compatible Computer

The Incredibly Quick Professional XT Computer

Sale Price SALE

**$449.95



List $1195 (Add $10.00 shipping.*)

- ***15 MHz (Norton CI Rating)**
- **Parallel, Serial & Game Ports**
- **TTL Monochrome Monitor Card**
- **512K Professional XT Turbo Computer**
- **Super Fast V20 CPU**
- **AT Style Keyboard**
- **Security Keylock**
- **Clock Calendar**

** Must be purchased with **MS DOS & G.W. BASIC** Software for an additional **$79.95.** (Not sold separately.)

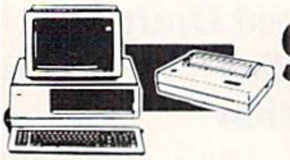
IBM® is the registered trademark of International Business Machines, Inc.

Complete *15MHz IBM® XT Compatible System

Computer, Drive, Monitor, Printer & Software

Sale

Sale Price



$659.95

Professional System List $2495 (Add $35.00 shipping.*)

No One Sells This System For Less!

- ***15 MHz (Norton CI Rating)**
- **512K Professional XT Turbo Computer**
  4.77/8 MHz ; Clock/Calendar
  Parallel, Serial & Game Ports
- **MS DOS 3.2 & G.W. Basic** Included
- **Hi-Res 12" Monochrome Monitor**
- **Genuine IBM 8½" Printer** with Interface And Heat Transfer Paper
- **Software:** Word Processor, Spreadsheet & Data Base

IBM® is the registered trademark of International Business Machines, Inc.

Complete Apple® Compatible System

Computer, Drive, Monitor, Printer & Software

Sale Price Sale

$479.95



List $1228 (Add $35.00 shipping.*)

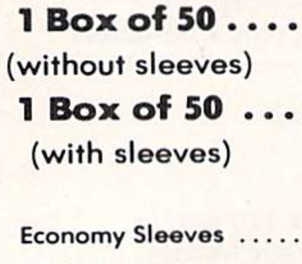
- **Laser 128K Computer**
  Video, Parallel & Serial Ports
  Runs Virtually All of the Apple II/IIc/IIe Software
- **Hi Res 12" Monochrome Monitor**
- **Genuine IBM 8½" Printer** With Interface And Heat Transfer Paper
- **Software:** Word Processor, Spreadsheet & Data Entry

No One Sells This System For Less!

Apple® is the registered trademark of Apple Computers Inc.

5¼" Floppy Disks Double-Sided, Double-Density

Each Disk 100% Certificated • Automatic Dust Remover
Free Lifetime Replacement Warranty



New Low Price!

19¢ Each

(Add $4.00 shipping.*)

1 Box of 50 .... $9.50 (19¢ each)
(without sleeves)
1 Box of 50 ...... $10.50 (21¢ each)
(with sleeves)

Economy Sleeves .... $1.00 per quantity of 50
Disk Labels................ $3.95 per quantity of 100

15 Day Home Trial • 90 Day Immediate Replacement Policy

COMPUTER DIRECT
22292 N. Pepper Rd, Barrington, IL. 60010
Call (312) 382-5050 To Order!

* Illinois residents add 6½% sales tax. All orders must be in U.S. Dollars. We ship to all points in the U.S., CANADA, PUERTO RICO, & APO-FPO. Please call for charges outside continental U.S. or C.O.D. **MAIL ORDERS** enclose cashier check, money order, or personal check. Allow 14 days delivery, 2 to 7 days for phone orders and 1 day express mail. Prices and availability subject to change without notice. Shipping and handling charges are not refundable. (Monitors only shipped in continental U.S.) **COMPUTER DIRECT** will match any valid nationally advertised delivered price on the exact same product with a comparable payment method (excluding any applicable sales taxes). A physical copy order of the current valid lower priced ad must be supplied with the order, or within 15 days of date of purchase. **VISA — MASTERCARD — C.O.D.**

Reader Service No. 124

MOVIE SCROLL

For the C-64

By Peter M.L. Lottrup

Have you ever wished you could smooth-scroll whatever text you wished across the screen? Wouldn't it be just great to professionally smooth-scroll help screens, or title screens of your own programs? Or have you ever wished you had an easy way to present the credits for all your home videos? With *Movie Scroll* you'll be able to give your programs the professional touch, and show off your C-64's capabilities.

Movie Scroll will let you scroll from bottom to top any text on your computer monitor. The text is defined in a BASIC text array variable, and you can set the colors and the scroll speed you wish. The program is very easy to use and to add to your own programs.

TYPING IT IN

The smooth-scrolling routine is presented as a BASIC loader. The routine start address is defined by variable SA of the loader program in line 10. To load the routine elsewhere in memory, simply change the value of SA, and SYS to the corresponding new start address. The routine is entirely relocatable in memory, so it should be very easy to use with your own programs. The first program on page 79 is the BASIC loader for the ML routine.

USING *MOVIE SCROLL*

Movie Scroll needs you to supply a couple of parameters before you call it. First of all, the text you want to display must be included in a string array. The array is displayed in order (if you call the routine starting with A$(1), the second text displayed will be A$(2), A$(3), etc.). The smooth scroll continues until the first empty string element is encountered (" "). Then the routine starts to display the first element once again.

Before you call the routine, you must ensure that that the last variable used is the first text to be displayed. This is easy. If you defined the text to be displayed in array A$, and the first element is A$(1), include the line

```
A(1)=MID$(A$(1),1)
```

just before the SYS to the scroll routine. This operation will leave the string unchanged, but will ensure that it was the last string operated upon.

To control the scrolling speed of the text, you must define two delay values. These values must be POKEd into memory at addresses 820 and 821. The fastest scroll is with a "1". The slowest is with "255" (or zero). Sample values are:

```
POKE 820,100 :POKE 821,100
POKE 820,45 :POKE 821,200
POKE 820,2 :POKE 821,2
```

You can experiment with your own values to find the speed that suits your needs.

A final parameter which must be specified is the color of the text to be scrolled. As the routine does not use a second text screen for the scroll, it is not fast enough to shift color memory along with text memory without jerking, so color memory is not shifted. In consideration of some older C-64 computers, color memory is filled by the ML routine with the color value found at memory location 250. POKE the color value you wish here, and the text will be displayed in that color. If you wish to create your own multicolor backgrounds and then call the scroll routine without having it change the background color (as in the demo prograam) call the routine with SYS SA+22, where SA is the normal starting address of the routine.

When these parameters have been POKEd to memory, you can call the routine with SYS SA, where SA was the starting address where the routine was loaded (do not use a variable; use the numeric address).

The scrolling routine is compatible with all character commands (except with color codes in the text—as color memory isn't shifted with the text, you'll lose the selected color in the text after the line is scrolled up the first character). The routine displays reverse characters, redefined characters, and others.

Pressing any key will exit the scrolling, with the keycode detected being returned in memory location 250. By PEEKing this address, you can decide what key was pressed, and your BASIC program can proceed accordingly.

You must remember that to scroll an empty line of text, you should make that array element equal to a space or other "invisible" character, e.g., A$(18)=" ". Otherwise, if it is an empty element, the routine will think that all strings have been scrolled and start over. This is the technique used in the demo program. The last string in the series of strings to be displayed should be set equal to a null string (" ").

Load and run the sample program Demo after running the BASIC loader, to get an idea of the potential of this short, relocatable routine. ☐

SEE PROGRAM LISTING ON PAGE 79

RUPERT REPORT

Have you ever used any of those exotic disk drive commands? Hasn't their rather cryptic appearance intrigued you? If you have never tried any of the direct access disk commands, here is a chance to do something different.

Did you know it is possible to recover a disk file which has been accidentally scratched? We will discuss and create an elementary but functional "file undeletion" program. We will look firsthand at the raw data on the disk and see how to modify it.

Normally a disk drive is a rather mundane device. You put diskettes into it to save or load programs and data. Nothing very exciting.

The Commodore computer disk drive is special. Each one is really a computer in itself, complete with microprocessor and memory. With the right know-how, you can put little programs into it. You may use its memory as a data buffer to be accessed by the main computer. This month we will begin an investigation into the inner workings of the 1541 and 1571 disk drives.

WARNING: DANGER AHEAD

An old expression says that a little bit of knowledge can be dangerous. That is particularly relevant to our work this month. We will present an overview of the more useful disk commands, including some of the direct access commands. Be forewarned that a mistyped number or a misplaced bit may do irreparable damage to any data stored on the diskettes you will use.

Consequently, put away *all* diskettes which contain anything valuable while you are becoming familiar with these commands. I recommend that you format two new disks. Use one of them to store this month's program. Label it "Good" and put a write-protect tab on it. Use the other one for testing. Don't be tempted to test your programs on the "Good" disk. I destroyed the directory structure of my test disk at least a dozen times while I developed this month's program. If something goes wrong and you lose an hour's worth of typing, don't say you weren't warned.

With the caveats behind us, we can get started. BASIC 2.0 on the C-64 explicitly implements only three disk drive commands: LOAD, SAVE, and VERIFY. BASIC 2.0 includes several other commands which follow this format:

```
1 OPEN filenum, 8, 15
2 PRINT# filenum, command$
3 CLOSE filenum
```

These instructions 1) open a "logical file" to the command channel (channel 15) of the disk drive (device number 8), 2) send a command to the drive, and 3) close the file.

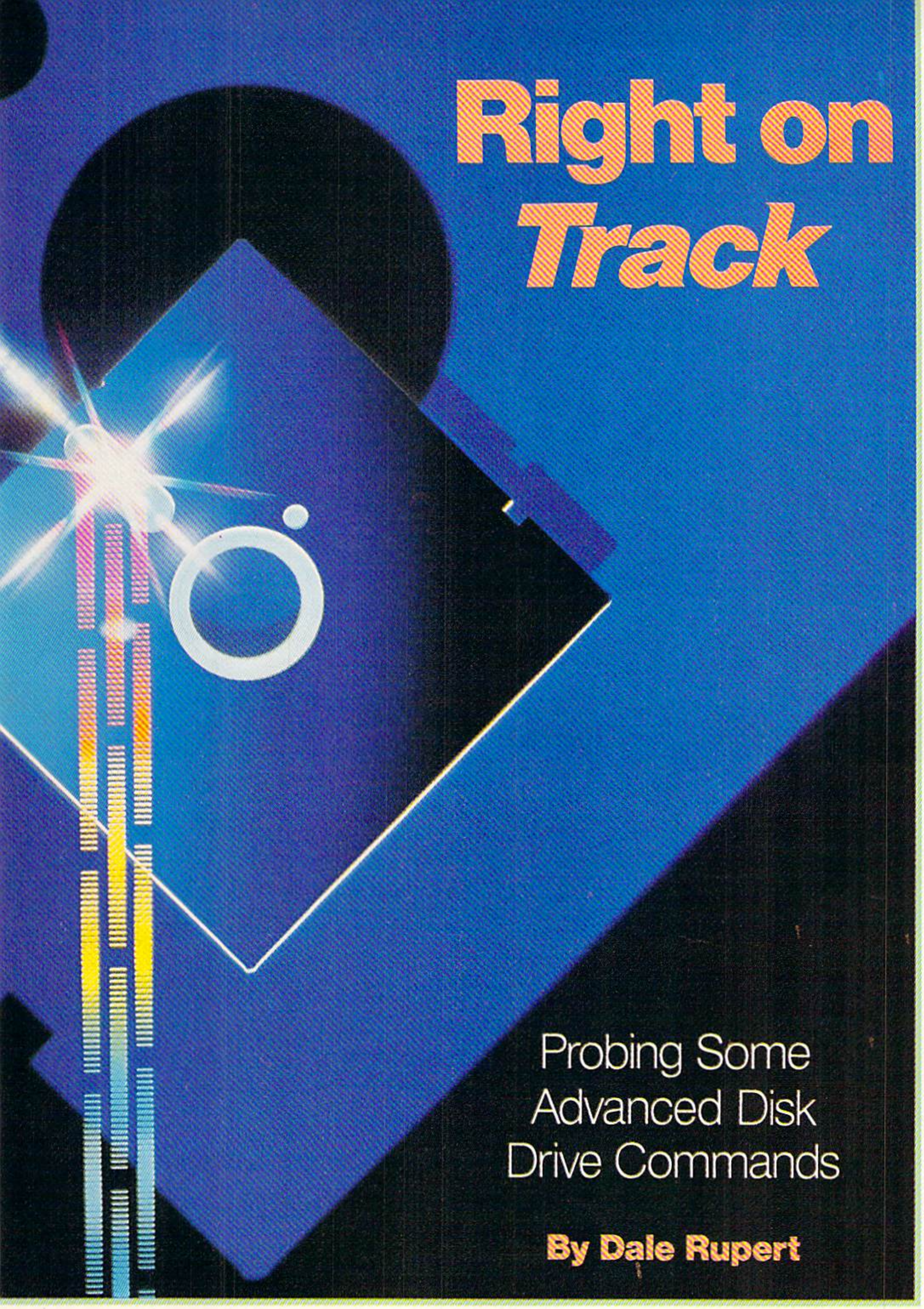
These three instructions are frequently condensed into two:



Right on *Track*

Probing Some Advanced Disk Drive Commands

By Dale Rupert



STEVEN HUNT / THE IMAGE BANK

```
1 OPEN filenum, 8, 15, command$
2 CLOSE filenum
```

The logical file number "filenum" can be any whole number from 1 to 127, although 15 is generally used to match the command channel number.

The "command$" may be any of these command strings:

```
"COPY: newfile = oldfile"
"INITIALIZE"
"NEW: diskname, id"
"RENAME: newname = oldname"
"SCRATCH: filename"
"VALIDATE"
```

The lowercase words are supplied by the user. The uppercase words are the commands. Note that it is not necessary to use the entire command name. Each may be abbreviated by its first letter. For example, to format a blank disk, you generally type something like this line in direct mode:

```
OPEN 15, 8, 15, "N:MYDISK,A1" : CLOSE 15
```

(Do not use this command unless you really want to erase everything currently on the diskette!)

Note that there are many options and additional parameters which may be added to some of the commands above (such as wildcards, drive number, and file type). You must study the User's Guide for your disk drive to cover all the possibilities.

BASIC 7.0 implements several new disk commands: DLOAD, DSAVE, DVERIFY, COPY, CONCAT, APPEND, SCRATCH, RENAME, and COLLECT, as well as the disk file commands DOPEN and DCLOSE.

FILES, CHANNELS, IDs, UGH

The computer can communicate with devices such as the disk drive, printer, modem, and screen by treating them as files. Analogous to the way you store and recover items within "physical files" in a filing cabinet, the computer transfers commands, programs, and data between its memory and these "logical files."

The file input/output commands can be quite complicated. The conglomeration of logical file number, device number, secondary address, channel number, drive number, and ID number is confusing, overwhelming, and downright ridiculous for most everyday operations. Figure 1 shows a simplified computer setup and may help clarify the difference between some of these parameters. In this figure, the computer has opened a link to the disk drive's Command Channel. The computer will communicate with the Command Channel with statements such as PRINT#14 and INPUT#14 since the OPEN statement gives the logical file number as 14.

Each peripheral attached to the computer has its own "Device Number." The Device Number for the disk drive is

Reader Service No. 116

8. Two drive systems typically use numbers 8 and 9. The printer and screen are other devices with numbers 4 and 3 respectively.

The Channel Number (sometimes called the "secondary address") is the number used by the microcomputer within the disk drive. Channel numbers 0, 1, and 15 are special: 0 and 1 are for the normal program loads and saves, and 15 is for the Command Channel.

To further complicate matters, there are several 256 byte buffers within the disk drive. One of them is dedicated to the Command Channel. Four others, numbered 0 through 3, can be assigned to a particular Data Channel. In Figure 1, we let the computer arbitrarily assign one of the buffers to Data Channel #2, since we usually don't care which one is used. The "#" in the OPEN statement could have been "#0" or "#3" for example if we wanted to select a certain buffer.

Only the numbers 8 and 15 in Figure 1 are predetermined. We may use other values for the logical file numbers as well as for the Data Channel number. Often we pick the same number for the logical file number and the channel number, just for simplicity.

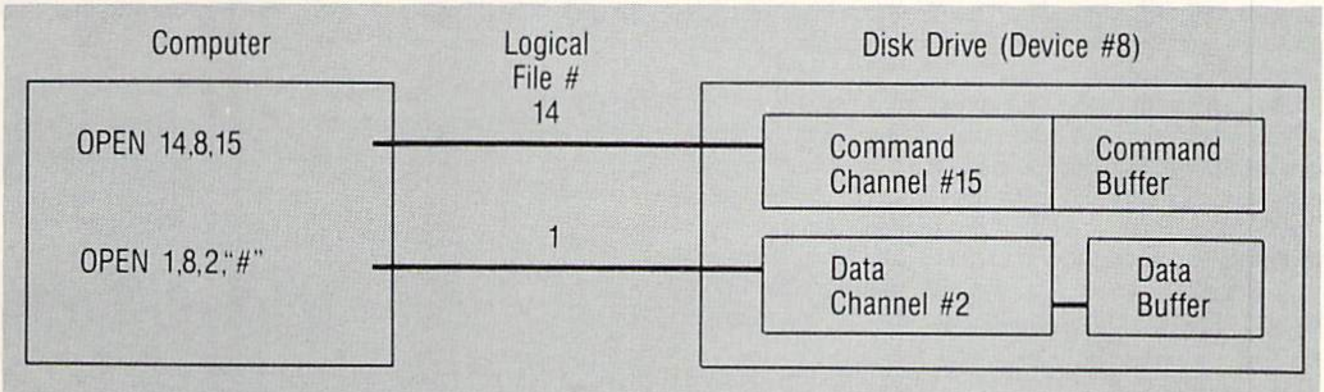


Figure 1—Computer/Disk Drive Communication

DIRECT DISK ACCESS

Normally we load and save programs and sequential files without concern for the data buffer within the disk drive. We specify the logical file number, the device number, the channel number, and the file name, and let the computer take care of the disk data buffer. Consequently we don't worry about how or where the data is stored on the disk.

It is possible with direct access commands to tell the disk drive exactly which track and sector we would like to read or write. The three commands we will use this month are "Block-Read," "Block-Write," and "Buffer-Pointer." These commands are sent from the computer to the Command Channel in the disk drive. They each refer to the data buffer associated with a specific Data Channel. Consequently, to use them we must open the Command Channel and open a Data Channel.

Here is a model for directly accessing Track 18, Sector 1 of the disk in drive (device) number 8:

```
1 OPEN 15,8,15
2 OPEN 1, 8, 2,"#"
3 PRINT# 15,"U1";2;8;18;1
4 GET# 1,D$
5 PRINT# 15,"B-P";2;0
6 PRINT# 1,D$;
7 PRINT# 15,"U2";2;8;18;1
8 INPUT# 15,EN,EM$,ET,ES
9 CLOSE 1 : CLOSE 15
```

There are "B-R" and "B-W" commands for block reads and writes, but they are not used for a variety of reasons. Instead, the new improved versions should be used: "U1" and "U2". As if the syntax for these commands were not abstruse enough (be careful of the commas and the semicolons), now the nicely mnemonic commands "B-R" and "B-W" are replaced by the more obscure "U1" and "U2". No one said these commands were for beginners.

Lines 1 and 2 above open the Command Channel and a Data Channel. Line 2 tells the disk drive to assign a disk memory buffer to Channel 2 just as we saw in Figure 1. Line 3 shows how the computer sends a "block read" command to the disk drive. Notice the logical file number of the Command Channel (15), the Data Channel number (2), and the Device number (8). The last two numbers specify the Track (18) and Sector (1) on the disk which is to be read into Channel 2's data buffer. Information about the first eight files in the directory is stored in Track 18, Sector 1 of every disk.

Each sector contains 256 bytes. Line 3 copies those bytes from the disk into the disk drive data buffer. Line 4 reads one byte from the disk buffer into a string variable D$ in the computer. At this point, D$ could be displayed or changed if desired. We won't do anything with D$ in this example.

Line 5 uses the "buffer-pointer" command to indicate which byte in the disk buffer we want to access. In this example we want to put D$ back into the buffer at its original position, so the "B-P" command sets the pointer to Channel 2's buffer to byte 0. We haven't changed byte 0 of the buffer, and we really do not have to put D$ back since the original value is still there. We will put it back anyway just to show how it is done.

Line 6 writes D$ from the computer back into Channel 2's buffer. D$ is put at the position indicated by the buffer pointer set in line 5, that is, at the first (zeroth) byte in the buffer. The semicolon after D$ is important so that a carriage return is not added by the PRINT# command.

Line 7 sends the "block-write" command to the disk drive's

command channel. This causes the disk drive to take the 256 bytes in Channel 2's buffer, and to write them to Track 18, Sector 1 of the diskette. Notice that the numerical parameters for the read and write commands are identical. Since the data in the buffer wasn't really changed, the data on the disk is not changed, although it is rewritten.

Reading from the Command Channel in line 8 returns the error status of the latest disk access. The error status is returned in four variables: the error number EN, the error message EM$, and the track ET and sector ES on which an error occurred (if any). It is desirable to read the error status on the Command Channel after each disk access. If an error occurs, the disk drive will sit with its light blinking until the error status is read.

Line 9 closes the Data Channel and then closes the Command Channel.

DIRECTORY SCANNING

The program *Directory Scanner* (see page 77) uses the sequence of commands in the model above to read all sectors of the diskette containing the directory. The first sector of the directory is at Track 18, Sector 1. The sectors of the directory are linked together. The first two bytes of each directory sector are the Track and Sector values of the next sector in the chain.

The main loop of the program is from line 170 to 280. After initialization and after opening the necessary channels, the program reads the first directory sector into the disk drive memory buffer in line 170. The subroutine at line 2000 reads the disk buffer into computer memory byte by byte. The next directory track and sector values are stored in TK$ and SC$. (Note that any CHR$(0) read by the GET# command is interpreted as the null string " " and must be converted into CHR$(0) as shown in line 2070.)

The remaining 254 bytes in the buffer are accumulated in B$. The bytes in B$ appear as shown in Chart 1 (recall that bytes 0 and 1 from the disk buffer are in TK$ and SC$).

Notice that each sector of the directory stores up to eight file names. Each file entry is 30 bytes long. The bytes used by the program *Directory Scanner* are as shown in Chart 2. Your *1541* or *1571 User's Guide* shows the complete Directory File format.

The routine at line 3000 picks out the relevant bytes within B$ and displays the filename, type, starting track and sector, and length of file in 256 byte blocks.

B$ Byte	Definition
1-30	File entry 1
33-62	File entry 2
65-94	File entry 3
97-126	File entry 4
129-158	File entry 5
161-190	File entry 6
193-222	File entry 7
225-254	File entry 8

Chart 1: Bytes in B$

Once the eight filenames for the first directory sector are displayed, the routine at line 4000 uses TK$ and SC$ to locate the next sectory of the directory if the user desires to continue. Since all directory entries are contained within Track 18, the end of the directory file is reached when the next TK$ does not give a value of 18.

If there are more directory entries, line 280 loops back to line 170 where the next directory sector is read into Channel 2's buffer and the process is repeated. If the user wants to quit, or if there are no more directory entries, line 4100 closes the channels and ends the program.

File Entry Byte	Definition
0	File type (0=DEL, 1=SEQ, 2=PRG, 3=USR, 4=REL) OR'd with 128 if properly closed; OR'd with 192 if locked
1-2	First Track and Sector of file
3-18	File name padded with CHR$(160)
28-29	Number of bytes in file (low-byte, high-byte integer format)

Chart 2: Bytes Used by *Directory Scanner*

UNDELETION MAGIC

You may notice when you run this program that there are filenames of numerous deleted files still in the directory. When a file is deleted, its file type in the directory is changed to 0. Also, the sectors it originally occupied are "freed up" in the Block Availability Map (BAM). The BAM keeps track of which sectors on disk are allocated and which ones are free. The actual data of the deleted file is not changed on the disk. The road map to find all of its pieces is obliterated.

The disk VALIDATE command takes the starting sector of each undeleted file and follows through the chain of links, ensuring that all sectors of a file are properly allocated. In order to undelete a file, all we need to do is to change its file type from 0 (deleted) to 128 (undeleted PRG file), then let the disk drive execute the VALIDATE command to put the pieces back together.

It is important to understand that this process is successful *only* if no other file has been saved since the deleted file was scratched. If another file has been saved in the meantime, quite possibly some or all of the deleted file's sectors have been recycled and now store the new file's data.

Consequently, *do not* use this program to undelete a file

unless you are sure that its deletion was the last operation performed on the disk. I have successfully recovered many deleted programs and sequential files with *Directory Scanner*. But the only time this program can work properly is immediately after scratching the file.

High-powered, commercial quality file undeleters will check the BAM for conflicts before they reestablish the scratched file's links. This program does not do that. This program is meant to show you how the undeletion process works. You will find it works well if you go back and reread and obey the previous paragraph.

You must delete the REM statements at the start of lines 230 and 260. They are there so you can debug your typing without any possibility of destroying the diskette. Once your typing is correct (the program runs without errors), delete those REM statements and save the program once more.

You can undelete any number of files on a disk, but you can only undelete the files within one block of eight within each directory sector. Press "U" to undelete a file. Press any other key to ignore the deleted file. Line 5090 changes the file type and status in B$ to PRG type (TYP=2) closed properly (OR'd with 128). You may modify this line to indicate a different file type, or replace the 128 with 192 to create a locked file.

Once the eight files within the current directory sector have been displayed, if you selected any of them for undeletion, the routine at line 6000 is called, which writes the updated file status data to the current sector and validates the disk. If all went well, line 6090 tells how many files were undeleted, and line 6100 ends the program. If you did not select any files within this directory sector to be undeleted, UDFLG (undelete flag) is false, and execution resumes at line 270.

You should practice with this program before you use it for the "real thing." On a newly formatted disk, save a short program a dozen times or so under different file names. Run *Directory Scanner* just to see the files listed.

Now delete a few of the files, then run the program again and undelete them. They should all reappear in their original form. It may be necessary to run the program more than once to undelete them all if they reside in separate directory sectors.

Since you are only doing this on a test disk for now, save a long program after deleting some of the short ones. Then try undeleting the short ones. Chances are the long program will have overwritten the sectors formerly used by the short files, and you will end up with a mess. At least you will have learned the limitations of this undeletion process which does not analyze the BAM before undeleting. Don't undelete if you have written to the disk since the deletion. At least copy all valuable files to another disk before you try.

In future columns, we will look further into the disk structure and the nature of the BAM. At that time, you will be able to write an undeletion program which will compete with the best of them. ☐

SEE PROGRAM LISTING ON PAGE 77

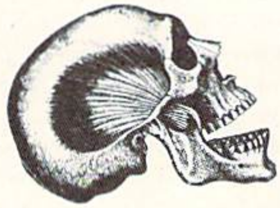
PROGRAMMING YOUR OWN TEXT GAMES



Part I of III: Create a New World

By Cleveland M. Blakemore







Something incredible has happened to twentieth century civilization. Literature has come to life. What was once a passive medium has now become an interactive one. The art of reading has lent itself to genuine experience, breathing life into surreal worlds.

This miracle has occurred through the wonderful medium of the electronic computer, a device almost everyone in our culture now has access to. Due to the advent of a piece of software called the text adventure, the reader can become a part of the story itself. The reader is now the protagonist and determines the outcome of the book, choosing from a variety of climaxes and conclusions that the writer/programmer has prepared in advance.

The problem seems to be that while everyone wants to know how to write text adventures, there is an inadequate dissemination of the information required for programming them. The three-part series of articles beginning this month is written for average to advanced BASIC programmers who are interested in creating and playing full-length text games that they have designed, either for their own pleasure or the enjoyment of others. The three programs accompanying the articles in this series range from tiny miniadventures to sprawling epics.

Most of the books I have seen on text games frequently use a very simple collage of print statements and GOTOs to replace good programming logic. This is not practical for large projects, and is a poor way to learn programming.

Using the modular approach described in this series, you should be able to create and debug a full-length adventure game in about 30 to 40 hours of programming time. Later, you will find that it lends itself easily to expansion or modifications.

All text games have a simple structure, the same for any game:

1. Get the player's input
2. Analyze his input
3. Take the correct action
4. If the player has not won then begin again at step one.

Since the structure and algorithm for every program is identical, you can start work on a new game at any time by just loading in an old game and hacking away at the basic framework.

THE IDEA MACHINE

"Where do you get all your ideas from?" is a question commonly asked of me as a programmer. I often feel as though I should ask in return, "How do you stop your ideas from driving you crazy?!?"

Every human being on earth is a natural dynamo of creative energy. Learning how to tap this energy and translate it to a book, a canvas, or a computer's memory, is a skill that can be learned. It is far from being a gift of the gods, or innate. It is more a state of mind, an openness to your own dreaming soul.

If you are reading this magazine, you have demonstrated an active mind. Now you must learn to twist your mental imagery, your dreams, secret places, and fantasies, into concrete lines of BASIC code.

THEMES

What interests you? Suspense stories? Sword and sorcery? Science fiction, science fact? Intrigue, horror, romance?

Believe it or not, all these themes are suitable for text adventures. All of them could probably be enjoyed by people of much wider or varied interests, if they were written in such a way as to convey the excitement of the author over the subject.

Colossal Caves is the grandfather of all text adventures. Written by Don Woods and Will Crowther at MIT on a SAIL PDP-10 mainframe back in the infancy of micros, it would prove to be the forerunner of everything from the *Zork* trilogy to currently popular games with far more advanced parsers.

The classic text game *African Adventure,* about a search through darkest Africa for Dr. Livingston, is a public domain adventure game that has delighted microcomputerists for years. It is available on the CompuServe network, as well as many other bulletin boards, and still manages to convey a sense of mystery and adventure years after its creation. Games like that are almost immortal. It will probably

still be floating around 50 years from now.

These games have made their authors some money through the years. But I suspect most of them were not written for profit, but from a deep creative urge on the part of the programmers. They had envisioned a world populated by fantastic creatures, wizards, and menaces, and could not rest until they had seen it in a form others could appreciate and enjoy.

If you need ideas for text games, look around you. Books. Movies. Your own dreams, nightmares.

Tolkienesque fantasy is enjoying a great deal of popularity. Tales of wizards and warriors are powerful myth-archetypes that reside in the universal subconscious of mankind. They strike a clear bell in our primitive psyches, playing on the elemental race memories of all humans. Some contend that it's because these images fade in the minds of people as they grow older that they lose their power; it is in young people that they are most likely to stir the mind and excite the imagination. One fact is clear—few things are more fascinating than a computer game with castles, dungeons, ogres, and treasure.

Science fiction allows a great deal of room for originality. It is a medium in which almost anything can happen. Marooned spaceships and interstellar junkyards make good plot devices for text adventures. Aliens make terrific allies, or horrifying enemies.

Suspense and Mystery require the author to stick to more concrete rules for the storyline, but the result can be much more challenging and subtle. Deadly spies hunting the player, hidden time bombs, or mysteriously murdered diplomats make for anxiety-ridden games.

Horror stories about mansions with ghosts and concealed fortunes, or subways with creepy crawling "things" in the shadows, have a neat flavor of their own. Again, the programmer can let his imagination run wild.

ONLY ONE WAY TO START...START!

The way to begin is to get three blank sheets of paper. Write the title and theme of the game at the top of Page 1. If it is simply too weird to label, write "Fantasy" beneath the title. Underneath this, describe the goal of gameplay. This can be as simple as "Get plenty of money and get out" or "Return the Sceptre of Knossos to the Castle Gallstone and save the princess from the evil Serpent God."

Now write a few short paragraphs describing the world in detail. Write down all vital locations. Give them names. Visualize each one as clearly as possible. Make the game world small and modest, with preferably five to ten locations for your first game.

On Page 2, write down all obstacles that will hinder or prevent the player from winning the game. These are the obstacles in the path of game completion. They can be as simple as getting the correct key for the correct door, or as difficult as lowering a reservoir's water level enough so that you can cross over to the other side. On the same page, write a list of every object that the player can carry in this game, up to 12. Give each object a number.

On Page 3, draw a map of the world. Label all locations with a name and a number. Now decide what objects go in which rooms, and draw the appropriate arrows.

These are all the notes you need to make a full-fledged text adventure, from beginning to end. We'll elaborate on each page as we work on our game, but we'll do it on the computer where it counts, not on paper. Because this series is mostly interested in the interactive use of BASIC as a programming medium, we'll keep the "dry computing" and "flowcharting" to a minimum. It's boring and futile, as you will undoubtedly make changes once you get on the machine.

Have patience, however. It's going to take a little work to get that world in your mind to sit up and act right within the computer.

But it's also worth it.

You'll have until next month to refine that vision on those three sheets of paper into a perfectly conceptualized storyline, when we'll begin walking through the step-by-step procedure of creating the code to breathe life into it.

To whet your taste buds and warm you up, *Lost Dutchman's Mine* is included in this issue. Type it in and enjoy it. It's simple, but don't worry if you can't understand it. Next month I'll demonstrate how I changed three pieces of paper, just like the ones you have, into a complete program. Also in Part II will be another program, slightly more advanced than this month's offering. The third and final installment of this series will offer a text adventure rivaling commercial programs in challenge and complexity.

Until then...dare to dream! It makes fools into great men, and great men into fools. ☐

SEE PROGRAM LISTING ON PAGE 80

MOB BLOB

Synchronized Sprite Movement on the C-64

By H. L. Morimodo

Commodore was wise to include hardware inside the C-64 for the management of sprite shapes. These Movable Object Blocks (MOBs) are probably the machine's most useful feature.

The Video Interface Chip handles the attributes of each of the eight sprites without any need for the user to load in software to supplement the computer's abilities. Sprites can have any of 16 colors, be expanded in either the X or Y axis, and move smoothly on or off screen. They even have collision detection registers to indicate when they make contact with other screen objects such as characters or bit map graphics.

Their only real limitation concerns their size. The resolution of all sprites is fixed at 24 by 21 pixels for all eternity, beyond the alteration of any programmer, no matter how clever. This can be very constraining for programs that need relatively larger objects. I have tried to surmount this hardware limitation with the short machine language interrupt program, *MOB Blob*.

The program moves sprite one around the screen under the control of a joystick in Port 2, and will move around any of the other eight shapes at distances which you can specify, relative to sprite one. This enables you to create and use images larger than one sprite shape by combining several into one large image under joystick control. The game possibilities with *MOB Blob* are numerous, depending on your ingenuity. You may find it useful for some utility applications as well.

The program defaults are set up so that a mere SYS command will produce eight sprites side by side, moving smoothly under the control of Port 2. The default screen limits keep the eight sprites within the borders, but you can change them to whatever you like.

If you have no sprites in memory, you can enter these three lines to set things up for a little experimentation:

```
POKE53248+21,255

FOR X = 0 TO 63:POKE 13*64+X,255:NEXT

FOR X = 0 TO 7:POKE2040+X,13:NEXT
```

These three short lines will give you a *MOB Blob* square to work with so that you can play around with the routine.

SYS49152 and the interrupt routine will wedge itself into the regular IRQ.

To change the left and right screen limits, use this SYS command:

```
"XSET"

SYS 49507, XR,.,XL
```

where XR is any number between 0 and 512, for the right border screen limit. XL is the left border limit and is assumed to be to the left side of the most significant bit line on the screen. The period is merely a dummy variable. It can be any number you wish. The entry point just requires it after getting the number XR.

If you'd like to set the upper and lower limits on the Y axis, use:

```
"YSET"

SYS 49532,YU,YL
```

where YU is the upper screen limit and YL is the lower. Make sure that YL is more than YU or you'll get screen flickering as the interrupt tries to keep your irrational human mind satisfied.

If you don't need all eight sprites included in the *MOB Blob* image, you can alter the default mask inside the interrupt to only move certain sprites at offsets to sprite one:

```
"BITEMASK"

SYS 49545,BM
```

where BM is equal to a number between 0 and 255, with

the appropriate Boolean bits set within the value to indicate which sprites you want included in the *MOB Blob*. Remember that sprite one has a value of 1, with each preceding sprite indicated by the next power of 2. If you'd like sprites 2, 3, and 4 included in the interrupt, then SYS 49545,2+4+8. To include all sprites in the routine again, SYS 49545,2+4+8+16+32+64+128.

Although the joystick control defaults to movement in all directions, you can control this movement with another entry point that masks out various stick directions:

```
"SETJOY"

SYS49552,SJ
```

To limit the image to just up and down movement, SYS 49552,17. To move it left and right on the horizontal plane only, SYS49552,68. For diagonals only, SYS 49552,170. For X and Y axis directions only, try SYS 49552,85. Finally, to set it back to eight directions, SYS 49552,255.

The two routines that follow are the heart of the *MOB Blob* control system: they set up the offset tables for the sprites to indicate at what distance from sprite one they will be moved in tandem under joystick control. The program defaults to all eight sprites side by side in two rows, but you might want to set up a long column, a row, or even an irregular shape inside your own program. Both SYS points are entered with the same format: the sprite number, from 0-7, in the first value, and the offset in pixels in the second.

```
"OXSET"
(Set X-axis offset from sprite one)
SYS 49559,SN,D

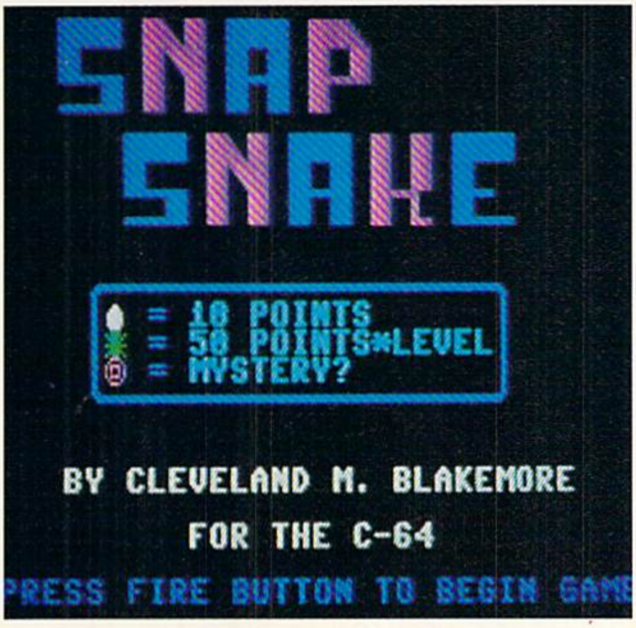
"OYSET"
(Set Y-axis offset from sprite one)
SYS 49576,SN,D
```

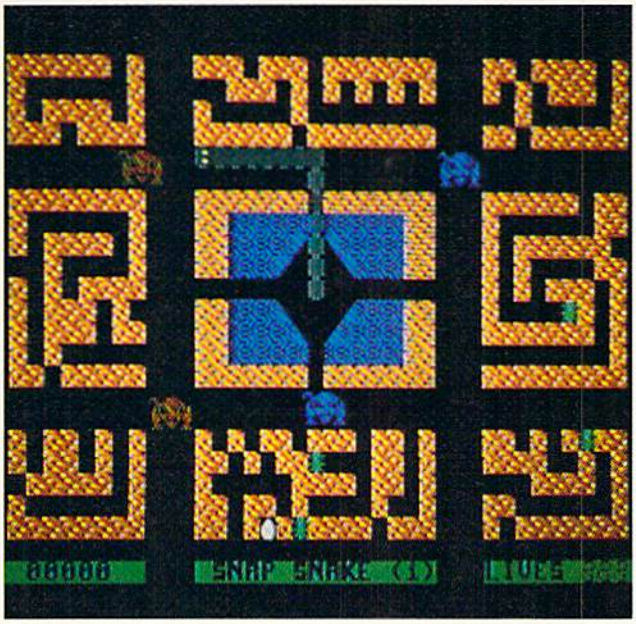
You can achieve a variety of effects by changing the default values here. You could create a ring of sprites, a square box, or any shape you care to design. A little hacking goes a long way here, so have patience and keep playing with it until you get it right.

Last but not least, you can set the speed with which the image moves by POKEing a number into 49625. The default is 1 here, 3 is really moving, and 6 is flying along. You can change this number any time you want to, before or after you SYS 49152 to start the interrupt routine.

The entry points do not perform error checking on the values you send them, so make sure you keep your values within the limits or you'll get either a syntax error, some weird effects from the interrupt, or (worst case) a complete system crash. That's the risk you run whenever you use machine language. □

SEE PROGRAM LISTING ON PAGE 74





PHOTOS: MICHAEL R. DAVILA

If you ever get a chance to visit Texas, be sure to ask about the "snap snake" native to those parts. You'll end up with some red-faced farmer swearing up and down that one of those critters whipped his great grandson fifty lashes on the back when he tried to run the serpent over with the tractor last year. But you'll notice that it's always somebody else–his nephew, his thrice-removed cousin. Ask him if he's ever seen one, and he'll deny ever actually laying eyes on this awesome species–but he'll take a blood oath to the effect that he's heard them cracking and slapping stones to powder far off in the fields on some hot afternoon.

According to the great unwritten herpetological almanac of Texas legend, these snakes are extremely dangerous creatures that curl themselves into a hoop by swallowing their own tails. Rolling like wheels, they obtain the necessary horizontal speed, then catapult themselves angrily at terrified humans, disengaging at the last possible second and delivering a mighty whiplike blow with their hind end that knocks the helpless recipient goofy.

When I was going to college in Texas, the locals related this story to me time and time again, the same story so often repeated that it sometimes seemed there might be a grain of truth to it.

In any case, I have never seen a snap snake. However, I was so impressed with this folk tale that I thought it would make a neat computer game, especially aurally. I was right. *Snap Snake* is one of my favorite games in terms of sound. I hope you'll find it entertaining as well.

Plug your joystick into Port 2 to play. Only vertical and horizontal controls are used to control the snap snake.

My snap snake does not roll into a hoop. His only resemblance to his legendary cousin is that he makes a snapping sound as he extends out from and back into his hole.

Since snakes are notorious egg-stealers, I set the game in the desert, with the snake's burrow in the middle of a maze of tunnels belonging to desert birds, who bury their eggs in the sand. The snake is in a good location for easy meals all day long, except for one thing. His natural enemies, possums, roam the area constantly. They are ravenously hungry as well. If they touch any part of his body, he will end up a meal himself.

Our snake is a wary customer, though. He keeps his tail end anchored in the middle of his lair, and when a possum approaches, he spins back and withdraws into his hole. Press the fire button with the joystick centered to pull the snake back into his burrow.

Besides the eggs, there are occasional bugs in the nests that the snap snake can swallow for bonus points. If he is really fortunate, he may stumble on a gold nugget, which can be worth anywhere from 200 to 1000 points.

As soon as the snap snake clears the screen of eggs, the game moves to a new level, with faster possums, more eggs, and different colored terrain. Level 9 is the highest obtainable, and it is very difficult to clear the screen at this level.

Snap Snake is a hybrid of BASIC 2.0 and machine language, utilizing interrupts to play the music and move the possums. It also has multicolor graphics so that I could make easy color changes to the screen with little effort. I stuck it up in video block 3 (49152-65535) so that I'd have plenty of room for the character definitions. Make sure that you type POKE648,4 if you hit RUN STOP/RESTORE accidentally during gameplay. You'll be unable to see the cursor until you do.

Also be cautious when typing in the egg locations at the end of the program. *Bug Repellent* might not catch a transposition error here, but you should spot the problem if any egg, bug, or nugget appears anywhere but at the end of a tunnel. It could cause false collision data to occur if it is in the path of one of the possums.

Snap Snake will probably remind you of some of those zany, addictive games that were released in droves throughout the early half of the '80s, such as *Centipede* or *Frogger.* I hope you'll find yourself spending many frustrated hours playing "just one more game" of *Snap Snake.* ☐

SEE PROGRAM LISTING ON PAGE 74

BACK ISSUES OF Ahoy!

#2—FEB. '84 Illustrated tour of the 1541! Artificial intelligence! Synapse's Ihor Wolosenko interviewed! String functions! And ready to enter: Music Maker Part II! Screen Manipulation! Night Attack! Relative Files!	**#3—MAR. '84** Anatomy of the 64! Printer interfacing for VIC & 64! Educational software series begins! VIC game buyer's guide! And ready to enter: Address Book! Space Lanes! Random Files on the 64! Dynamic Power!	**#5—MAY '84** Future of Commodore! Inside BASIC storage! Memory management on the VIC & 64! Guide to spreadsheets! Hurray for arrays! And ready to enter: Math Master! Air Assault! Biorhythms! VIC Calculator!	**#7—JULY '84** MSD dual disk drive! Database buyer's guide! Training your cursor! Screen displays! File Sleuth! Users Groups! And ready to enter: Renumbering! Checklist! Math Defender! Brisk! What's My Job?
#8—AUG. '84 Choosing a word processor! Computational wizardry! Creating your own word games! Sound on the 64! And ready to enter: Micro-Minder! Directory Assistance! The Terrible Twins! Words Worth!	**#9—SEPT. '84** Program your own text adventure! Build a C-64 cassette interface! Video RAM! Word processors, part II! And ready to enter: Salvage Diver! DOS! Sound Explorer! The Castle of Darkness! Base Conversions!	**#10—OCT. '84** C-64 graphics programs! Bit-mapped graphics! Joystick programming! Graphics processing! And ready to enter: VIC 40 Column Operating System! BAM Read & Print! Emerald Elephant! Lawn Job!	**#11—NOV. '84** Music programs & keyboards for the 64! Graphics feature continues! 2-D arrays! And ready to enter: PTE word processor! Block Editor! Alternate Character Set for the 64! Tunnel of Tomachon!
#12—DEC. '84 Buyer's guide to printers! 1525 printer tutorial! Fast graphics with custom characters! User Guide to KMMM Pascal! Diving into BASIC! And ready to enter: Construction Co.! Space Patrol! Cross Ref!	**#13—JAN. '85** VIC and 64 OS exposed! Sprites! Insert a 1541 device # disconnect switch! Ghostbusters! And ready to enter: Ultra Mail! Music Tutor! Alice in Adventureland! Midprint! To the Top! Tape/Disk Transfer!	**#14—FEB. '85** Printer interfacing! Multicolor sprites! Modems! Bulletin Boards! Theory of game design! Buying a modem! And ready to enter: Futurewar! Fontasia! VIC Eraser! Insurance Agent! Flankspeed! Telelink 64!	**#15—MAR. '85** Creating multiscreen gameboards! Inside the Plus/4! Commodore DOS! And ready to enter: Old Routine! Programmable Functions! Automatic Line Nos.! Home Budget! Salmon Run! Numerology!
#16—APR. '85 Assembly language column begins! Programming the joystick! 1541 disk drive alternatives! The Kernal! And ready to enter: Hop Around! Faster 64! Booter! Elecheck! BASIC Trace! Space Hunt!	**#17—MAY '85** Disk drive enhancements! Install a reset switch! Assembler escapades! And ready to enter: Super Duper! Two-Column Directory! DSKDU! Raid! DOS Plus! Font Editor! Tile Time! Interrupt Wedge!	**#18—JUNE '85** Music & graphics entry system! How modems work! Inside the 6510! And ready to enter: Quad-Print! Mapping 4.4! Towers of Hanoi! Speedy! Duck Shoot! Bit Dumping! Screen Magic! 6510 Simulator!	**#19—JULY '85** PROM Programming! 3-part harmonies on VIC/64! Speeding pixels! And ready to enter: Auto-Append! Script Analysis! Wizard of Im! Lucky Lottery! Brainframe! Etch! Printat! Autos: Leasing v. Buying!
#20—AUG. '85 Inside the 128! Real-world simulations! Sound effects! Modems! And ready to enter: Windows! Formatter! Sound-a-Rama! Screen Dump! Selectachrome! Disintegrator! Fidgits! Gators N Snakes!	**#21—SEP. '85** Inside the 1571 drive and 128 keyboard! Sprite programming! And ready to enter: Fastnew! Go-lister! File Lock! Dragon Type! Superhero! Auto-Gen! Moxey's Porch! Fish Math! Ahoy!Dock! Invective!	**#22—OCT. '85** Create cartoon characters! Infinitesimal intrigue! Secrets of copy protection! And ready to enter: Shotgun! Maestro! Solitaire! Mystery at Mycroft Mews! Gravinauts! 1541 Cleaning Utility! Shadey Dump!	**#23—NOV. '85** Guide to adventure gaming! ML sprite manipulation! BASIC for beginners! And ready to enter: Lightning Loader! Knight's Tour! Chopper Flight! Rhythmic Bits! Instant Bug Repellent! File Scout! Slither!
#24—DEC. '85 Speech synthesizer! The IBM Connection! The year's 25 best entertainments! And ready to enter: Gypsy Starship! Directory Manipulator! Cloak! Gameloader! Jewel Quest! Lineout! Santa's Busy Day!	**#25—JAN. '86** Build a speech synthesizer! Survey of sports games! And ready to enter: Martian Monsters! Streamer Font! Microsim! Haunted Castle! Knockout! Infraraid! Alarm Clock! Memory Check! Scratch Pad!	**#26—FEB. '86** Windows! Build an autoexec cartridge! Align your 1541! Survey of flight simulators! Structured programming! And ready to enter: Arena! Head to Head! Crabfight! Treasure Wheel! Character Dump!	**#27—MAR. '86** Programming educational games! Memory dumpers! Choosing a copy program! Custom characters! And ready to enter: Ahoy!Term 128! Trivia Game Maker! Brickbusters! Easy Lister! Programmer's Aid!
#28—APR. '86 Comet catching! Survey of action and strategy games! Screen dumping! And ready to enter: Chrono-Wedge! Mr. Mysto! Air Rescue! Notemaker! Screen Window! JCALC! Hidden Cavern! Swoop!	**#29—MAY '86** 128 graphic bit map! Epyx strategy guide! 128 commands! ML music programming! And ready to enter: Bigprint! Star Search! Failsafe! English Darts! Ski Folly! Free RAM Check! Alchemist's Apprentice!	**#30—JUNE '86** Debugging dilemmas! Public domain software! Winning at Ultima! Computer Aided Design! And ready to enter: LazyBASIC! Got A Match? Star Strike! Queen's and Bishop's Tours! Shaker! Trackdown!	**#31—JULY '86** Inside the Amiga! Conditional branching! Chess programs! 128 and 64 DOS! And ready to enter: Screen Sleuth! Skull Castle! Head-on! Nebergall Run! Wordcount! Crazy Joe! Fidgits' Music School!
#32—AUG. '86 Inside the Amiga, part II! Approaching infinity! C-64 war simulations! Pascal for beginners! ML graphics! And ready to enter: Reversi! Highlight! Disk Cataloger! Meteor Run! Trim! Step On It! Flap!	**#33—SEPT. '86** Windows and viewports! Sound & music on the 64! COMAL! And ready to enter: The Last Ninja! Speech64! Multi RAM! Dogcatcher! Trapped! Matchblocks! Variable Manager! Dual Dump! Mine Canyon!	**#34—OCT. '86** Build a digital oscilloscope! ML speed techniques! And ready to enter: Vault of Terror! Quick Change! Penguins! Attack Force! Disk Checkup! Dvorak Keyboard! Mountaineer Mack! 128 to 64 Autoboot!	**#35—NOV. '86** C-128 shadow registers! Data file handling! PROMAL! Habitat! And ready to enter: Teleporter! 128 RAM Check! Discs of Daedalus! Guardian! Tenpins! Syntax Patrol! Deluxe List! Long Lines! Detonation!
#36—DEC. '86 File manipulation! C-128 shadow registers! Football games! And ready to enter: The Artist! Minotaur Maze! Mouse in the House! Lazy Source Code! Rebels and Lords! Speedway! The Editor! Micro City!	**#37—JAN. '87** Pointers and the monitor! Best games of '86! DOS for beginners! And ready to enter: Vortex! Hanger 14! BASIC Ahoy! Catacombs! Lixter! Dark Fortress! PermaLine! Starfighter! Bugout! Screens!	**#38—FEB '87** Hacking into machine language utilities! Amiga RAM expanders! And ready to enter: Window Magic! Crunchman! User Conventions! The Adventurer! More BASIC 128! Jailbreak! Turtle Rescue! 640!	**#39—MAR. '87** Basic esthetics! Survey of video digitizers! Multiplayer games! And ready to enter: C-64 Compressor! Wizard Tag! Turbopoke! Rescue 128! Lights Out! Pinball Arcade! Stow Away! Caverns of Geehonk!
#40—APR. '87 Inside the Amiga 2000! Fractals! Baseball games! COMAL, turtle graphics, and Logo! And ready to enter: Infoflow! Laps! Pieman! List Formatter! Scrambler! Extended Background Mode! Planet Duel!	**#41—MAY '87** Kernal power! 64 and Amiga graphics! Microworlds in COMAL! Brain games! Dark Fortress master maps! And ready to enter: Moondog! Startup! Illusion Master! Wall Crawler! Scavenger Hunt!	**#42—JUNE '87** Megaflops and microseconds! Sci-fi braingames! C-64 to Amiga file transfer! And ready to enter: D-Snap! Wraiths! Galactic Cab Co.! Cave of the Ice Ape! ALT-Key 128! Power Squares! 128 Multi RAM!	**#43—JULY '87** Real world interfacing! Bit map graphics tutorial! C-64 graphic conversion! Martial arts software! And ready to enter: Wizard Tag II! Data Creator! Plink & Plonk! Univaders! Data Express! 128 Scroller!
#44—AUG. '87 Electronic screen swapping on the C-128! Science fiction action games! The death of GOTO! Amiga reviews! And ready to enter: Archer! Banner Print! Route 64! Steeplechase! Batter Up! Scanner!	**#45—SEP. '87** The 128 versus The Clones! Building an Amiga trackball! MSD update! CES Report! And ready to enter: Crystallus! Spriteshell! Hoops! Chainmail! No News! PS to GEOS! Centerfold! Red Alert!	**#46—OCT. '87** A rainbow of data structures! BASIC 8.0 reviewed! Buying guide to COMAL! Tips Ahoy! Art Gallery! And ready to enter: Empire! C-64 RAMDrive! Hotfoot! Platforms! Spray-Cam! Jam Attack!	**#47—NOV '87** The ins and outs of data organization! Overview of C-64 memory expanders! Commodares! And ready to enter: Orbit! RAMCO! A-Maze-Ing! Line Sentry! Desert Front! Paper Route! Flash Flood!
#48—DEC. '87 Exploring artificial intelligence! Expansion port tutorial! Memory expanders! And ready to enter: Redirect! Silhouette! Fueling Station! Take Two! Pizza Boy! Sprite-On! Warship! Cliffhanger!	**#49—JAN. '88** Iterative mathematics and graphics! Expansion port tutorial, part II! The best C-64 and Amiga games of 1987! And ready to enter: Infoflow 64! TextSave! Clone-A-Matic! Ice Titans! Jungle Jake!	**#50—FEB '88** Investigating audible wave phenomena! Cartridges! Commodore 64 emulators for the Amiga! And ready to enter: Coffee Break! Crypt of Fear! Screen Wizard! ML Ranger! 128 Smart Merge! Marauder!	**#51—MAR '88** Synthesizing waveforms! Comprehensive guide to Commodore power supplies! Utilizing Amiga HAM mode! Art Gallery! And ready to enter: Vee Kloros! The Extractor! 3-D Graphic Projector! Phantasy!
#52—APR. '88 BASIC's hidden treasures! Updating your Amiga 1000's printer port! Disk drive accelerators! And ready for to enter: Time Subway! IRQ Messenger for the 64 and 128! Sprite Exploder! Hi-Res Windows! Barricade!			**#53—MAY '88** Premier issue of *Ahoy!'s AmigaUser*, featuring: survey of sampling software! The essential Amiga game library! Guide to Amiga books! Business column! And ready to enter: AmigaUserTerm! Matrix Pattern!

Please send me

____ copies of issue #____ ____ copies of issue #____

____ copies of issue #____ ____ copies of issue #____

Enclosed please find my check or money order for $________
(outside the USA add $1.00 per copy)

NAME______________________________

ADDRESS____________________________

CITY________________STATE______ZIP________

Send to: *Ahoy!* Back Issues, Ion International Inc., 45 West 34th Street—Suite 500, New York, NY 10001.

If ordering more than four issues, list choices on a separate sheet.

All issues $4.00 except #2, #3, and #21 ($10.00)

TIPS AHOY!

Contributors to *Tips Ahoy!* will be compensated at highly competitive industry rates. Send your best programming and hardware hints for the C-64, C-128, or Amiga to *Tips Ahoy!*, c/o Ion International Inc., 45 W. 34th St.—Suite 500, New York, NY 10001. Include a stamped and self-addressed envelope if you want your submissions returned.

Compiled by Michael R. Davila

PRINT 24/40 ERASE 20/40

Try to place a character on the bottom line in the right hand corner and the whole screen scrolls up one line, right? Not on my Commodore 64, at least not since I put together my PRINT24/40 routine. Of course I then found that I needed an erase routine that would handle the same area. These routines control line 24, position 40 without disturbing anything that has already been printed on the screen in any other place.

You simply type in these routines anyplace you like in your own program and then create your own A$. You must, of course GOSUB to the line that you have chosen for the PRINT24/40 routine to print your A$. To erase the bottom line just GOSUB to the ERASE24/40 routine and it's gone.

Line 15 contains a time delay just so that you can see how these routines work. In your own programs you may continue some other screen activity until you want to erase, and then go to the ERASE24/40 subroutine. Hence, both the time delay and the END may be taken out or otherwise adjusted to suit.

—Robert Ridout
Roswell, NM

```
5 PRINTCHR$(147)"[9"[DOWN]"][6"[RIGHT]"]
 PRINT 24/40 [4"*"] ERASE 24/40":REM FOR
 DEMO ONLY
10 A$="IS THIS THE BOTTOM LINE OR WHAT?"
15 GOSUB2000:FORW=0TO1000:NEXT:GOSUB3000
 :END
2000 B$=LEFT$(A$,LEN(A$)-2)+RIGHT$(A$,1)
2005 POKE214,24:POKE211,(40-LEN(A$)):SYS
 58732
2010 PRINTB$+CHR$(157)CHR$(148)+MID$(A$,
 LEN(A$)-1,1)+CHR$(19):RETURN
3000 POKE214,24:POKE211,0:SYS58732
3005 PRINT"[39" "]"CHR$(20)CHR$(32)CHR$(
 19)
3010 RETURN
```

NEVER TOO MANY CLOSE'S

When OPENing a logical file, whether it be OPEN4,4 for a printer or OPEN2,8,2 for a disk file, always precede the OPEN with a CLOSE command, like so:

```
1000 CLOSE2: OPEN2,8,2,"DISK FILE,S,R"
```

This never hurts, and avoids the infamous ?FILE OPEN error if you forgot to close the file previously. Note that in Commodore BASIC a file doesn't have to have been OPENed for you to CLOSE it. Try this:

```
1000 FOR L=1 TO 128:PRINT L
1010 OPEN 1,8,15
1020 NEXT L
```

It'll crash every time! Whereas this works:

```
1000 FOR L=1 TO 128:PRINT L
1010 CLOSE 1
1020 NEXT L
```

While interesting, that last one doesn't really do anything—it's just a little demonstration. However, if you modify line 1010 like so:

```
1010 CLOSE L
```

then you have something potentially useful. Every logical file OPENed previously will be closed (well, every logical file numbered less than 129). Note that this will close *all* logical files, even files opened to the printer or modem. If you wish to shut down just the logical files associated with a disk drive, do this:

```
CLOSE 15: OPEN 15,8,15: CLOSE 15
```

Every file on the disk drive, whether it be opened for read, write, issuing commands, or direct-access, will be closed safely—no "splat" (*PRG, *SEQ, etc.) files will be created. I often include this line in the start of my programs just to be extra cautious.

—David F. Paulsen
Houston, TX

C-64 PROGRAM PROTECTOR

While using my sector editor, I discovered an interesting way to protect programs from being accidentally scratched. The directory portion of the disk (Track 18, Sector 1) contains various information about the programs contained therein. Below is an example of the sector editor's output of Track 18, Sector 1:

```
00: 00 FF 82 11 00 4D 59 20 .....MY
08: 50 52 4F 47 52 41 4D A0 PROGRAM
10: A0 A0 A0 A0 A0 00 00 00      ...
18: 00 00 00 00 00 00 01 00 ........
```

The third byte ($82) is the file type. $81 signifies a sequential file, $82 a program file, $83 a user file, and $84 a relative file. To make your program file safe from accidental scratching, simply change the $82 in the third byte to $C2. After this value has been changed, the program file cannot be scratched by ordinary means. This excludes reinitializing the disk using the NEW statement. It should also be noted that this works only for program files (PRG or $82). If you

attempt to change anything other than a program file, all data will be lost.

—John Thelen
Caledonia, WI

BLOCK DISPLAYER

I often use a disk editor when examining file storage and disk format, but sometimes loading and running a disk editor is not worth the time.

With *Block Displayer*, it's a snap to examine individual tracks and sectors. And you have the option of seeing the data on paper, which makes further investigating easier. When viewing from the screen, the space bar serves as a pause feature.

—Shawn K. Smith
Bronx, NY

```
100 REM* BLOCK DISPLAYER -SHAWN K. SMITH
110 H$="0123456789ABCDEF":F$=CHR$(18)+":
120 INPUT"OUTPUT TO PRINTER (Y/N)";A$
130 OPEN15,8,15:OPEN8,8,8,"#"
140 DV=4+(A$<>"Y"):OPENDV,DV
150 INPUT "TRACK, SECTOR (DEC)";T,S
160 GOSUB230:FORD=1TO32:S$=F$:GOSUB270
170 FORT=1TO8:GET#8,B$:A=ASC(B$+CHR$(0))
180 GOSUB300:IFA>127THENA=AAND128
190 IFA>95ORA<32ORA=34THENA=46
200 S$=S$+CHR$(A):PRINT#DV,D$" ";:C=C+1
210 NEXT:PRINT#DV,""S$:NEXT:C=0
220 GOTO260
230 PRINT#15,"U1:";8;0;T;S
240 INPUT#15,A,B$,C,D:IFA=0THENRETURN
250 PRINTA,B$,C;D
260 PRINT#DV:CLOSEDV:CLOSE8:CLOSE15:END
270 A=C:GOSUB300:PRINT#DV,"00"D$": ";
280 GETA$:IFA$=""THENRETURN
290 GETA$:ON 1+(A$<>"")GOTO290:RETURN
300 H=INT(A/16):L=A-H*16
310 D$=MID$(H$,H+1,1)+MID$(H$,L+1,1)
320 RETURN
```

SWAP SPEED

Lately, I have been doing most of my programming with the Commodore 128 in 80 columns because working in FAST mode really does speed things up. But there are some things which I prefer to perform from 40 columns. So I must press ESC-X to switch screens, *and* issue the SLOW command so the 40-column screen is visible. Then to return to 80 column, it's the same boring process.

With *Swap Speed*, all that is needed is ESC-X to switch screen and operating speed. So if you're using the 80-column screen in fast mode, pressing ESC-X will not only switch the screen to 40 columns as it normally does, but it will also place the computer into SLOW mode. And pressing ESC-X again takes you back to 80 columns with FAST mode in effect.

—Shawn K. Smith
Bronx, NY

```
100 REM* SWAP SPEED BY SHAWN K. SMITH
110 S=5120:REM FOR C-128 FROM AHOY!
120 FORD=STOS+29:READY:POKED,Y:NEXT
130 A=INT((S+11)/256):POKES+6,A
140 POKES+1,S+11-256*A
145 POKED,PEEK(824):POKED+1,PEEK(825)
150 SYS(S):PRINT"SWAP SPEED READY.":END
160 DATA 169,011,141,056,003,169,020,141
170 DATA 057,003,096,201,088,208,014,036
180 DATA 215,048,005,032,182,119,208,003
190 DATA 032,199,119,169,088,076
```

ML PRINT

A lot of ML techniques to print strings to the screen utilize loops in conjunction with CHROUT. While this will certainly do the trick, there is another way that is almost as easy to use as the BASIC PRINT command.

Looking into the BASIC interpreter we find a routine labeled STROUT located at $ABIE. This is the routine actually used by PRINT. If we store a message at a location and then load the .A register with the low byte of this address and the .Y register with the high byte, we can JSR to STROUT and our message will be printed to the screen.

The string's maximum length is limited to 255 characters. These can be embedded cursor and/or color controls, CHR$ codes, etc. Also, the string must be terminated with a .BYTE 0.

This sample program will demonstrate the routine's use in practice.

—B. McQuirt
Columbus, OH

```
10 SYS 700
20 STROUT = $AB1E
30 * = $0CC3
40 ;
50 LDA #<MESSAGE
60 LDY #>MESSAGE
70 JSR STROUT
80 RTS
90 ;
100 MESSAGE .ASC "UP TO 255 CHARACTERS"
.BYTE 0
```

C-128 80 COLUMN SCREEN SAVE AND LOAD

The following routine will allow you to access the VDC memory for the 80 column screen so that you can perform a screen save or load.

First the bottom of BASIC is moved up by issuing a GRAPHIC1 command. This will provide a protected 8K area to which we can move the screen and attribute memory of the VDC chip. Then the ML portion of the program is POKEd to memory starting at 0B00 (the cassette buffer).

To load a screen, simply BLOAD your file into this protected area and then SYS DEC ("0B2F") to move it to the VDC memory. To save a screen, reverse the process. Move the information to 8192 and then BSAVE the file.

—Barbara H. Schulak
Pepper Pike, OH

```
10 REM C-128 80 COL. SCREEN SAVE & LOAD
20 :
30 GRAPHIC1,1:GRAPHIC5,1
```

```
40 :
50 FORI=0TO93:READA$:POKEDEC("0B00")+I,D
EC(A$):NEXT
60 DATA A9,00,85,FB,A9,20,85,FC,A9,00,8D
,5E,0B,A2,12,A9
70 DATA 00,20,CC,CD,E8,20,CC,CD,A0,00,20
,D8,CD,91,FB,C8
80 DATA D0,F8,E6,FC,EE,5E,0B,AD,5E,0B,C9
,10,D0,EA,60,A9
90 DATA 00,85,FB,A9,20,85,FC,A9,00,8D,5E
,0B,A2,12,A9,00
100 DATA 20,CC,CD,E8,20,CC,CD,A0,00,B1,F
B,20,CA,CD,C8,D0
110 DATA F8,EE,5E,0B,E6,FC,AD,5E,0B,C9,1
0,D0,EA,60
120 :
130 F$="FILENAME"
140 REM GOSUB1000 TO LOAD SCREEN
150 REM GOSUB1040 TO SAVE SCREEN
160 REM REST OF PROGRAM HERE
170 END
180 :
1000 BLOAD(F$),B0,P8192
1010 SYSDEC("0B2F")
1020 RETURN
1030 :
1040 SYSDEC("0B00")
1050 SCRATCH(F$)
1060 BSAVE(F$),B0,P8192TOP12288
1070 RETURN
```

C-64 RAM READER

Have you ever wished that you could read the 16K RAM underlying BASIC and KERNAL ROM just as easily as you can store values there? Storing a value there requires a simple POKE, but a PEEK to the same address yields the value stored in ROM, not in RAM. The following short routine is written to reside anywhere in memory. Just change the variable SA to the address where you want the routine located. Then, the USR function will act as a PEEK, with the difference that it will only read RAM. Use the function just as the PEEK function, entering USR followed by the address from where to read enclosed in brackets (expressions and variables allowed). For example:

```
10 POKE 63000,1
20 PRINT "ROM=";PEEK(63000)
30 PRINT "RAM=";USR(63000)
```

or,

```
A=USR(62000+B*8)
```

The following BASIC loader will install the routine in memory and activate the USR routine vector for you.

—Peter M.L. Lottrup
Buenos Aires, Argentina

```
10 SA=828
20 FORI=SATOSA+35:READA:POKEI,A:CK=CK+A:
NEXT
30 IF CK<>4252 THEN PRINT"ERROR IN DATA"
:STOP
40 POKE786,SA/256:POKE785,SA-(PEEK(786)*
256)
50 DATA 32,247,183,165,100,133,252,165
60 DATA 101,133,251,160,0,120,165,1
70 DATA 41,252,133,1,177,251,168,165
80 DATA 1,9,3,133,1,88,169,0
90 DATA 32,145,179,96
```

SING A STRING OF SID

The C-64 may be missing a convenient PLAY command like the C-128, but this short subroutine can be used to create an endless variety of themes using very little space in your program.

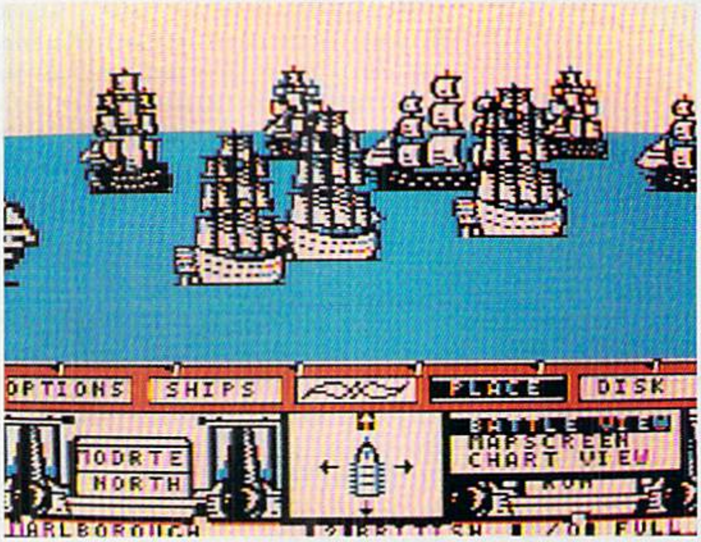
It accepts the legal alphanumeric characters from 33 to 95 ("!" to "[back arrow]") to specify an ascending range of tones with durations, in two character sets. The second character is a valid number 0 through 9 to indicate duration. For example, "A1B2" will play two tones, the second higher than the first and twice as long. I created the two sample songs in the demo in about five minutes, experimenting with different letter and duration combinations. Enter the subroutine with the musical data in the spring M$ and listen to the music play. The volume is faded gently at the end of each song before it returns to the line number it was called from.

—Cleve Blakemore

```
•10 REM PLAY COMMAND FOR C-64
•20 REM SET UP MUSIC NOTE ARRAY
•30 DIM H(107),L(107):S=54272
•40 PRINT"WAIT[3"."]CREATING NOTE ARRAY"
•50 FORX=.TO62:M=2703*(2[UPARROW]((X-20)/
12)):IFM>65535THENM=65535
•60 H(X)=INT(M/256):L(X)=M-H(X)*256:NEXT
•70 PRINT"PERILS AND DANGER[4"."]"
•80 M$="A6C6G6A6C6G6A4C4G4C8G8D8E6C6G6D6E
6":GOSUB140
•90 PRINT"THE LIGHT OF A NEW DAY[4"."]"
•100 M$="Q2S2L3L2Q2S3L1M1N1O2S2Q1R2P2Y6":
GOSUB140
•110 END
•120 REM SID STRINGSINGER SUBROUTINE
•130 REM NOTES AND DURATIONS IN M$
•140 FORX=.TO24:POKES+X,.:NEXT:POKES+5,4:
POKES+12,4:POKES+19,4:POKES+6,241:POKES+
13,241
•150 POKES+20,241:POKES+4,33:POKES+11,33:
POKES+19,33:POKES+24,15
•160 FORN=1TOLEN(M$)STEP2:M=ASC(MID$(M$,N
,1))-33:D=VAL(MID$(M$,N+1,1))
•170 POKES+1,H(M):POKES,L(M):POKES+8,H(M+
3):POKES+7,L(M+3)
•180 POKES+15,H(M+12):POKES+14,L(M+12):FO
RX=.TOD*100:NEXT:NEXT
•190 FORN=15TO.STEP-1:POKES+24,N:FORX=.TO
50:NEXT:NEXT:RETURN
```



Playing engineer is the most appealing aspect of The Train. *Simple commands access the throttle, furnace, steam blowoff, brake, etc.* *READER SERVICE NO. 112*



High Seas *is more suited to military simulation fans than to the casual gamer. Even a short game requires great concentration.* *READER SERVICE NO. 113*



M.I.S.L. Soccer *keeps more than half the field visible at any point. The screen scrolls as the ball carrier moves across the field.* *READER SERVICE NO. 114*

Featured This Month:

THE TRAIN (ESCAPE TO NORMANDY)

Accolade
Commodore 64
Disk; $29.95

The Train (Escape to Normandy) is one of those great ideas that falls completely apart somewhere between inspiration and realization.

Here's the story. It's August 1944, and the race is on between the advancing Allies and the fleeing Germans. Although the war in Europe is almost over, the retreating troops are anxious to take some souvenirs back to the Fatherland with them.

One such collection of keepsakes is a trainload of Picassos, Renoir, Monet, and Gauguin art objects parked at a depot in Metz. This priceless treasure trove is scheduled to head east for Berlin.

The player is cast as Pierre LeFeu, the leader of a group of Resistance fighters. These forces for France must seize the train and head west to the border to rendezvous with a group of Allied soldiers at dawn.

This harrowing scenario certainly contains the pathos and verve needed to excite all art lovers, friends of France, and right-minded saviors of the universe. The problem, however, lies in the execution. Unfortunately, the dramatic concept is pared down to a series of boring, uninspired action games. These are rendered even less appetizing through drab graphics and unsatisfying play mechanics.

The game boils down to three tasks: driving the train and participating in a pair of target games called Taking the Train/Taking the Station and Taking a Bridge.

Playing engineer is actually the most appealing thing about this simulation. Simple joystick commands access the throttle, furnace, brake, forward/reverse, steam blowoff, whistle, and cab signal indicators. In response to the appropriate signals, the Resistance handles the details of track switching.

Taking the Train is the weakest of the three scenarios. Unfortunately, this is

CHIP LEVEL DESIGNS PRESENTS

128 mode & CP/M version now available!

- RapiDOS -

THE SUPER-FAST PARALLEL DISK OPERATING SYSTEM FOR THE COMMODORE 64 AND 1541 DISK DRIVE!

- **All** disk access is handled at super-fast parallel speed! (LOAD, SAVE, directory, SEQ & REL files, scratch, validate, format).
- Designed to support multiple drive systems.
- Parallel Centronics printer support with file spooling capability.
- Many useful, timesaving features (DOS Wedge, screen dump, resident monitor).
- No loss of compatibility.
- Far too many features to list in this ad . . . and perhaps in this magazine! (Call or write to get all the details!)

. . . and if you want the ultimate, get RapiDOS Professional!

- Gives even faster disk access!
- Uses 8k RAM track buffering and hardware GCR conversion!
- Provides 40 track extension (749 blocks free!)
- Adds 20 new disk commands (i.e., lock files, change disk name).

Here's what people are saying about RapiDOS:

Mike J. Henry (Basement Boys Software) - *"It's amazing how incredibly fast it is, I'm impressed!"*
Mitch S. (Eaglesoft Inc.) - *"Very fast, very reliable, and very compatible. I love it!"*
J.F. Jones (ADP) - *"Superbases' speed is increased greatly, and it's now a dream to use!"*

Function	Normal DOS	RapiDOS	RapiDOS Pro	Your System
Load 202 blocks	128 sec.	15 sec.	3 sec.	
Save 202 blocks	196 sec.	98 sec.	8 sec.	
Format 35 tracks	90 sec.	24 sec.	18 sec.	

Compare these speeds with your current system and see why RapiDOS puts the C-64 into a different league!

RapiDOS requires a socketed kernal ROM U4, and is available in versions for the 64c, 128 in 64 mode, and 1541c (please specify when ordering). RapiDOS is easily upgradeable to the Professional Version. RapiDOS Professional drive controller is **(c) '87 mts data GbR,** the creators of the best European parallel systems.

At these affordable prices no C-64 owner should be without RapiDOS!

RapiDOS $49.95 **RapiDOS Professional $99.95**

MASS DUPLICATOR 1541

- For the C64/128 with a single 1541 disk drive.
- 15 second, 4 pass backup for standard disks!
- 25 second full GCR Super Nibbler, the most powerful yet!
- 9 second disk format!
- Fast loader!
- Quick installation. **$32.95**
- Can be upgraded to RapiDOS

MASS DUPLICATOR MSD

- A must for any MSD SD-2 owner.
- 15 second standard disk backup!
- 18 second full GCR Quick Nibbler!
- 9 second disk format! **$25.95**

MSD AUTO COPY ROM

- Adds new Fast Backup commands!
- Turns the MSD SD-2 into a dedicated copying drive (no computer needed). **$29.95**

C-64 BURST-ROM

- Gives the 64 'Burst Mode' when used with a 1571 or 1581 disk drive!
- Loads 100 blocks in 6 seconds on a 1571, 4 seconds on a 1581!
- Fast directory, SEQ, and REL file access!
- Built in DOS wedge!
- Simple installation. (Kernal ROM U4 must be socketed) **$32.95**

C-128 BURST-ROM

- Lets the 128 run at 'Burst' speed when in 64 mode!
- Provides the same features as the C-64 Burst-ROM! **$38.95**

STILL TO COME

- MSD SD-1 Fast Copier!
- 1571 Mass Duplicator!
- 1581 Utility Pack!

TURBO 64

- Speed-up cartridge for the C-64, just plug it in!
- Adjust the clock rate from 100 khz (1/10 normal) to 4 mhz (4x normal)!
- Uses 8/16 bit 65816 microprocessor (same as the Apple][gs... but twice as fast)!
- Spread sheets, BASIC, flight sims, graphics, and now GEOS... all are accelerated! **$189.95**

CLD

CHIP LEVEL DESIGNS

MasterCard Cash, Check, Money Order, M.C. or Visa VISA
$3.00 shipping on all orders
C.O.D.'s add $3.00

P.O. BOX 603
ASTORIA, OR 97103-0603

(503) 861-1622

Dealer, Distributor, & Group Pricing Available

the first challenge to be completed before anything more exciting happens. To successfully take the train, the user must provide covering fire for a comrade making his way across the train yard to throw the track switch. German sentries occupy a guard house that overlooks the railway. When a guard house window lights up, the soldiers within fire a burst of bullets. These travel slowly across the yard toward the player-surrogate, like a swarm of heavily sedated bees. If the gamer gets off an accurate shot to the illuminated window before the flock of bullets reaches him, the German gunfire disappears. (This encounter is not exactly the ultimate in realism.)

If the player can't fire a well-aimed round in time, he can duck the incoming fire by hitting the spacebar. However, this doesn't accomplish the Resistance's objective, since the comrade being covered won't continue his movement toward the switch under these conditions.

The Taking the Train scenario is repeated as a Taking the Station minigame when the player opts to occupy a station en route to meeting the Allies.

Once a station is free of enemy activity, the telegraph can be used to send a message to the Resistance, indicating a need for repairs. The Resistance then sends a reply and the train has two hours to reach the designated station for the repairs to be made.

The third minigame is the Taking a Bridge scenario. Here the player mans the train's flatcar-mounted cannon and attempts to destroy the enemy's gunship before it fires off seven rounds.

The graphics on all three games are heavy on grays and feature minimal animation. The dark ambience fails to invoke the spirit of these brave French fighters. Worse, the drab, sluggish environment does nothing at all to enhance the bland and frustrating game experiences.

The Train (Escape to Normany) is an interesting idea gone wrong. Instead of high adventure on the rails, it offers an unappetizing blend of simulation and symbolism that starts off on the wrong track and never really picks up steam along the way.

Accolade, 20863 Stevens Creek Boulevard, Suite E, Cupertino, CA 95014 (phone: 408-446-5757).

—Bill Kunkel

HIGH SEAS
Garde Games
Commodore 64
Disk; $49.95

The same qualities which make the Age of Fighting Sail so attractive to gamers hamper attempts to present this type of naval warfare on the computer. It is relatively simple to construct a statistical model of firing ranges and wind direction, but reproducing the romance and color of this bygone era is another matter entirely.

High Seas has all the intricate detail any armchair admiral could demand, but its lively presentation distinguishes it from a flotilla of accurate, but bland sea war games. The 11 scenarios included on the double-sided disk allow one or two commanders to test their strategic skill at anything from a meeting engagement between one vessel on each side to large actions with up to 15 ships. In addition, extensive ship- and scenario-creation routines enable the user to simulate any real or hypothetical battle from the period.

Ralph Bosson, who created *Under Fire* and *Tactical Armor Command* (both published by Avalon Hill), has always displayed an independent spirit and a freedom from design preconceptions. This makes him a perfect choice to author a game on a subject which stubbornly resists orthodox simulation techniques.

High Seas, like Bosson's previous works, is a radical departure from existing naval warfare programs. Most computer war games differ little from military boardgames. In Bosson's capable hands, however, the flexibility of high-speed data manipulation and electronic display creates a much more personally involving game environment than is possible with cardboard maps and symbolic counters.

Instead of presenting every phase of the engagement from directly overhead, *High Seas* gets right down to the waterline when the cannons boom. Aerial perspective facilitates ship movement and makes it easy to see the relative positioning of all vessels in multiship fights, but first-person viewpoint conveys the excitement of shot and shell much more effectively.

The drawings of the ships, particularly in the combat segment, look authentic. The multimasted ships are a tremendous improvement over the abstract icons used to identify units in most other war games.

The lack of detail in the deep blue backdrop undercuts the realistic ambience, but at least it does provide good visual contrast when the user scans the screen for enemy vessels. Unfortunately, memory limitations force *High Seas* to leave the smoke and fire of combat under sail in the players' minds instead of representing the "fog of war" on the screen.

A joystick-activated system of pull-down menus configures the parameters of the scenario and controls the activities of the ships during play. Moving the stick from side to side highlights portions of the menu bar which horizontally divides the screen between the active visual display and the status summary. Pressing the action button activates the lit menu.

Toggling the joystick forward and back permits the user to cycle through the available choices. Sub-menus, controlled with the same system, drop down into view as needed.

Despite its disarmingly charming graphics, *High Seas* is probably more suitable for lovers of military simulations than casual computer gamers. Though scenarios involving two ships can be completed in 15 minutes, even a short game requires tremendous concentration and some understanding of the tactical nuances of naval warfare.

The sheer size of its 44-page rulebook might daunt some novice admirals. Fortunately, writer Michael LePage has organized the information in a logical manner, so it's easy to look up hazy points during a game without wasting time. A tutorial keyed to Scenario 1 (Constitution versus Insurgente) is tremendously helpful for learning the routine of play.

Commanding elegant, though lethal, sailing ships of the line is a stimulat-

Reviewed Next Month:

- **Stealth Mission**
- **Express Raider**
- **Wooden Ships & Iron Men**
- **Plasmatron**

ing change from the tanks and bombers of World War II games. *High Seas* proves that naval warfare can be involving and exciting as well as mentally challenging.

Garde Games, 8 Bishop Lane, Madison, CT 06443 (phone: 203-245-9089).

–Arnie Katz

M.I.S.L. SOCCER
Mindscape
Commodore 64
Disk; $34.95

M.I.S.L. Soccer gives indoor soccer fans their shot at the ball in a computer simulation designed by Ed *(Superstar Ice Hockey)* Ringler. The computerist, as general manager, coach and/or on-the-field player, controls the game. There's not too much to kick about, either. Outstanding graphics, well-conceived play action, and smooth joystick handling make this an outstanding action-sports simulation.

Indoor Soccer, a modern hybrid of soccer and ice hockey, was invented in the United States during the 1970s. Two six-player teams (each with three forwards, two defenders, and a goaltender) play the game in hockey-sized rinks on artificial turf. Indoor soccer features most of the elements of traditional soccer, including passing, shooting, dribbling, headers, and bicycle kicks. These action moves are intensified by the smaller playing surface and a high plexiglass perimeter that keeps the ball from leaving the field.

M.I.S.L. Soccer brings much of the excitement from the playing field to the computer screen. Every phase of the game is adapted into a lively entertainment for sports fans, who can take or leave alone each of its components.

The GM recruits and trades players, reviews the team's history, and improves the team (via training camp). This is done through an allotment of points assigned at the end of each season. The first place team receives the fewest points, and the weakest squad gets the maximum number. This helps even the teams out a bit to make competition more keen in the new season.

In his capacity as coach, the user determines the lineup, then deals with substitutions when players tire or sustain injuries. The coach is also responsible for the overall strategy of the team.

Finally, the the user can don the cleats and, via the joystick, take his position on the field as either a midfielder or goaltender.

The center-forward, goaltender, and coaching duties can be assigned entirely to the computer or divided among two joysticks and the computer. This allows a pair of human players to team up against an entirely computer-directed squad as a player-coach or midfielder-goalie tandem.

The coaching area contains *M.I.S.L. Soccer's* primary weakness. Ringler attempted to one-up his earlier game with the introduction of player-by-player substitutions, in place of a line-oriented assignment system. This certainly *sounds* like an improvement, but the result is a game that bogs down in endless tactical decisions.

Every time play stops, the substitution process begins, grinding the game to a virtual halt while the coaches laboriously replace tired or injured team members one by one. This method is more meaningful in a sport like baseball, where most gamers are familiar with the individual players. Here, however, the names will be meaningless to the vast majority of users, so it turns into an exercise in studying qualifications. Fortunately, players are easily evaluated through a look at their "skill points," age, and, in the case of non-goalies, their "Actual Power' (AP) rating. These helpful stats are listed on the Substitution Screen.

The excellent graphics keep slightly more than half the field visible at any point. The screen scrolls smoothly as the ball-carrier moves horizontally across the field. The players are nicely animated and easily controlled by joystick, so midfielders can dribble, pass, and shoot, as well as deliver headers and bicycle kicks. Goaltenders can dive to the left or right. The recovery time after each save is, unrealistically, determined by the goalie's age.

But the gameplay is more than compensation for the design's minor flaws. *M.I.S.L. Soccer* is a fulfilling sports experience that lends itself to as much or as little depth as the user desires. Ed Ringler and his co-creators at DesignStar Consultants continue to establish themselves as masters of action-strategy sports simulations.

Mindscape, 3444 Dundee Road, Northbook, IL 60062 (phone: 312-480-7667).

Bill Kunkel

The publisher cannot assume responsibility for errors in the above listing.

Free Spirit Software Inc.

UTILITIES

ULTRA DOS UTILITIES
Module I

High Speed Hard Drive or dual floppy drive backup utility for the Amiga 500, 1000 or 2000. 512K Amiga required. Compatible with any hard drive that follows conventional AmigaDOS protocol. Backup those valuable files on your Hard Disk the easy way for only **$59.95!**

OXFORD PASCAL 128

OXFORD PASCAL 128 is an implementation of standard Pascal designed specifically for the C128. It offers all the enhancements of this powerful language together with some useful enhancements for the C128. Only **$39.95!**

SUPER AIDE

All-purpose utility program for the C64 provides:

- Bi-directional scrolling
- Auto Line Deletion
- Trace function
- Disassembler
- Lo-Res Screen Dump
- Number conversion (10, hex, binary)
- Append files
- Format — short new/complete new
- Menu-driven
- Auto Line Numbering
- Renumber
- ML Monitor
- List all variables to screen
- Hi-Res Screen Dump
- Restore newed Basic program
- Change Device number
- Packed Line Editor
- Determine file load address
- Change THIS TO THAT — search for all instances of specified string and replace with second specified string

• And much, much more!

Super Aide, the complete programmer's tool kit. Only **$29.95!**

". . . excellent, efficient program that can help you save both money and downtime."

Compute!'s Gazette
Dec., 1987

1541/1571 DRIVE ALIGNMENT

1541/1571 Drive Alignment reports the alignment condition of the disk drive as you perform adjustments. On screen help is available while the program is running. Includes features for speed adjustment. Complete instruction manual on aligning both 1541 and 1571 drives. Even includes instructions on how to load alignment program when nothing else will load! Works on the C64, SX64, C128 in either 64 or 128 mode, 1541, 1571 in either 1541 or 1571 mode! Autoboots to all modes. Second drive fully supported. Program disk, calibration disk and instruction manual only **$34.95!**

SUPER 81 UTILITIES

Super 81 Utilities is a complete utilities package for the 1581 disk drive and C128 computer. Among the many Super 81 Utilities features are:

- Copy whole disks from 1541 or 1571 format to 1581 partitions.
- Copy 1541 or 1571 files to 1581 disks
- Backup 1581 disks or files with 1 or 2 1581's
- Supplied on both 3½" and 5¼" diskettes so that it will load on either the 1571 or 1581 drive.
- Perform many CP/M and MS-DOS utility functions
- Perform numerous DOS functions such as rename a disk, rename a file, scratch or unscratch files, lock or unlock files, create auto-boot and much more!

Super 81 Utilities uses an option window to display all choices available at any given time. A full featured disk utilities system for the 1581 for only **$39.95!**

RAMDOS
Lightning Fast
RAM-DISK

RAMDOS is a complete RAM based "Disk" Operating System for the Commodore 1700 and 1750 RAM expansion modules which turns all or part of the expansion memory into a lightning fast RAM-DISK. RAMDOS behaves similar to a much faster 1541 or 1571 floppy disk except that the data is held in expansion RAM and not on disk. Under RAMDOS, a 50K program can be loaded in ½ second. Programs and files can be transferred to and from disk with a single command. RAMDOS is available for only **$39.95!**

GAMES

MONSTER POWER

BIG WHEEL MONSTER arcade action for the C64! One to four players can compete in Tractor Pulls, Mud Bogs and Monster Trucks.

Monster Power is only **$14.95!**

STRATEGIC PLAYGROUND FOOTBALL

Enjoy a nice game of football on the C64. Using playground rules, one or two players can compete. Get a treat from the icecream truck at half time.

Strategic Playground Football is only **$9.95!**



SUPER BIKE

Action-packed, fun-filled motor cycle arcade game for the C64. Race the clock in Motocross, Enduro, Supercross or Trials. Fly through the air on spectacular jumps. Bounce over woop-de-doos.

Avoid logs, trees, water holes, brick walls, other bikers, etc. as you vie for the gold cup.

Thrilling **Super Bike** action for only **$14.95!**

GALACTIC FRONTIER

Exciting space exploration game for the C64. Search for life forms among the 200 billion stars in our galaxy. Scientifically accurate. Awesome graphics! For the serious student of astronomy or the casual explorer who wants to boldly go where no man has gone before.

Only **$29.95!**

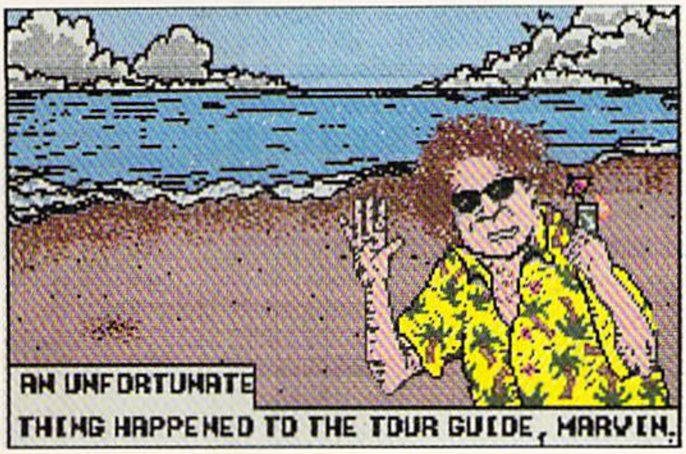
Order with check, money order, VISA, MasterCard, COD. Free shipping & handling on US, Canadian, APO,FPO orders. COD & Foreign orders add $4.00

MasterCard VISA

Order From: **Free Spirit Software, Inc.**
905 W. Hillgrove, Suite 6
LaGrange, IL 60525
(312) 352-7323
1-800-552-6777
For Technical Assistance call: (312)352-7335

SELECTED PRODUCTS AVAILABLE FROM FINANCIAL SYSTEMS SOFTWARE LTD IN ENGLAND 0905-611-463

Reader Service No. 144



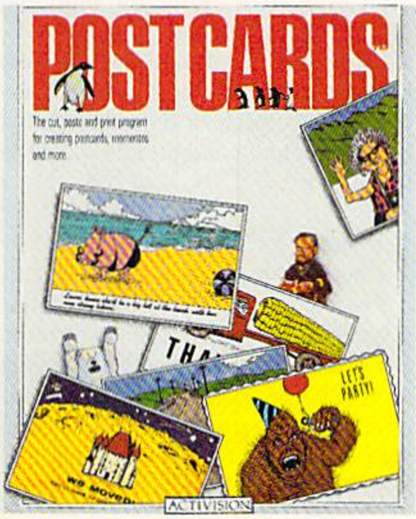
While most (we didn't say all) *of us don't take our 64's along on vacation, you can send customized* Postcards *by planning ahead.*
READER SERVICE NO. 130

POSTCARDS
Activision
Commodore 64
Disk; $24.95

Looking for a way to impress your friends with your computer? *Postcards* is a new twist in creativity programs. It gives Commodorians a chance to create their own customized postcards, using a delightful collection of clip art and backdrops.

It's a snap to put together a completely individualized postcard with this easy to use program. Using joystick or keyboard commands, first select a background from the 10 scenic paintings included on the disk. The wide range of geographic backdrops include Arctic, Beach, Cityscape, Country Road, Desert, Farm, Main Street, Moonscape, Mountain, and Volcano scenes.

Next choose art from the extensive files. There are 11 structures, ranging from the Arc de Triomphe to the Taj Mahal, a dozen road signs (Caution, Soft Shoulders, Speed Limit, etc.), a clutch of objects, over a dozen edibles, 28 people, 9 transportation vehicles, and more than 40 animals. Just choose the item(s) to illustrate your postcard and position them against the backdrop. Flip the pictures either vertically or horizontally, if desired, then paste them down. Next add your message–either take one from the small bank of phrases in the program or write your own. Finally, print out the finished creation, trim it down to size, and glue it to the postcard stock that comes with the program. Activision thoughtfully included



a glue stick in the package.

The whimsical art can be combined in hundreds, even thousands, of amusing scenes. However, *Postcards* also contains a simple paint program. There are 5 brush sizes, a selection of fill patterns, a line command for straight-edge work, zoom to make detail or touchup work easier, and of course an eraser to correct mistakes. Undo-previous-command rescues a picture from mistakes, and the Wipe command clears part or all of the screen. Artists can create their own from scratch, or customize the pictures from the clip file, so the number of designs available is as limitless as the imagination of the user. Postcards, whether finished or in progress, can be saved for reuse later.

The bank of phrases is very limited, but that's no problem since most people would rather write their own message anyhow. The small collection of cute captions is included just to get the user started. Messages are fun to compose, since there's a choice of alphabet fonts on the disk.

The program comes with a supply of preprinted postcard stock, and a format to prepare more with "place stamp here." There's also an order form for additional blank cards with the stamp block, but most people will probably use cardstock from their local stationers when the original supply runs out.

Postcards is fun to use, and the finished products are attractive and usable for dozens of occasions, as cards, invitations, memos, or even small signs. It takes only minutes to create something clever and unique.

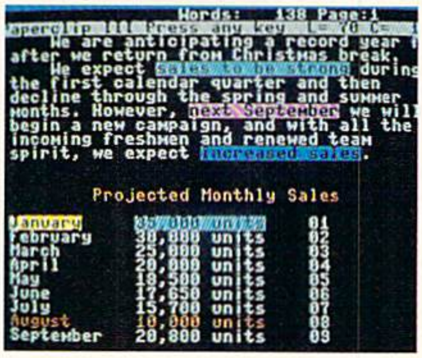
Activision, Inc., 2350 Bayshore Parkway, Mountain View, CA 94043 (phone: 415-960-0410).

–Joyce Worley

PAPERCLIP III
Batteries Included
Commodore 64 and 128
Disk; $49.95

PaperClip III is to the Commodore 64 and 128 what *WordPerfect* is to the IBM PC and the Amiga–the ultimate word processor. Both are powerful, cost effective, and relatively easy to use and learn.

This word processor is an extension and upgrade to its best-selling predecessors, *PaperClip* and *PaperClip II.* One disk contains the 64 version, the other contains the 128 code. The flip side of each contains a 40,000 word dictionary, a spell checker, and numerous printer files.



PC III: *state of the art 8-bit WP.*
READER SERVICE NO. 131

PC III, unlike its earlier incarnation, does not require a dongle (joyport security key) to operate. On the contrary, the disks contain a copy program to help you make working backups! Though the intention was good, the copy routine requires almost half an hour to duplicate a disk (one side). Take my advice—use a commercial copy program and spend the extra 40 or 45 minutes learning the program. That's about all it takes to understand the basics and a few of the advanced features.

Split Personality

The manual interleaves information about each version of the program. Commodore 128 specifics are boldfaced or otherwise highlighted so they stand out from the generic information. This arrangement works just fine, as the manual is thorough, well-organized, and complete.

C-64 load instructions are the usual run of the mill; however, the 128 version does not autoboot. Instead you must depress SHIFT and RUN simultaneously. That is my last gripe about either version of the program. From here on out it only gets better.

Other advantages of the 128 version include non-scrolling 80 column text entry and editing (with the proper monitor), automatic use of the 1750 RAM expansion module, optional interactive spell checking with extra RAM installed, and use of the 1571 disk drive burst mode to speed disk access. It's nice to see a program utilize a computer's unique characteristics.

Of course both sets of cursor keys are supported. So are the function keys, to perform tasks immediately (page down, page up, etc.) or to display menus (F1 to change screen parameters, F7 to view the main menu, etc.). There's even a somewhat unique function available: the ability to unformat text. It's intended primarily to strip RETURNs from downloaded text; the feature is also useful for stripping user-created documents or those transferred by a hardwired configuration over the serial port.

The Same but..

PaperClip has always had great cursor movement control. *PC III* continues the tradition. The screen format is likewise similar. A status line at the top of the screen displays any special conditions (insert mode, for example), and cursor position by row and column. (See the sample screen reproduced at the bottom of page 53.)

Above this is a command line which appears whenever the CTRL key is pressed. Pressing another key enters a command, such as reading a disk directory, performing searches, printing, and file handling. They are fully explained in the manual, including an excellent reference section.

Departing from former iterations, *PC III* also supports pull-down menus without the use of a mouse. Hit a function key and a menu drops. Run the highlight up or down via a cursor key and press RETURN to activate the selection. Some menu options allow setting changes—once highlighted, press the left or right cursor key to change the value. After that it's RETURN to activate.

For example, to change device numbers you would press F1 twice, cursor down to the Disk Drive Device option, and cycle through the options (4 to 15) with the left or right cursor key.

Tipping the Scale

PaperClip III weighs in, not just as a contender, but as the heavyweight champion. It would be difficult to find as feature-laden a word processor on any other 8 bit system.

Text can be inserted, deleted, copied, moved, ranged, searched, and replaced. Columns may be moved, shifted, repeated, deleted, erased (entries), sorted, or acted upon mathematically (with up to 38 decimal digit accuracy). Wildcard search and replace functions for single characters, complete words, and phrases are also supported.

A simple boilerplate capability is introduced via the "instant phrase" capability. Up to 52 one-line phrases can be tied to a keystroke and reproduced at will just by pressing the proper key at the desired time.

On the hard copy side, *PC III* handles proportional spacing, produces near letter quality dot matrix output, and provides 8 different pitches and pauses for paper insertion when using single feed sheets. (Your printer must be able to handle the commands.)

There are printers with options not supported by *PaperClip III.* To take advantage of these unique functions users can send "escape" or "control" sequence commands directly to the printer. Examples of this type include envelope and special sheet feed commands. Review your printer's manual for these options and the commands needed to drive them.

In the rare instance where your printer is not included in the list of drivers, *PC III* explains how to define a custom printer. In a similar vein is the character editor, for creating custom fonts and special display characters.

All previous *PaperClip* files can be loaded into *PC III.* So can files from Timeworks' *WordWriter 3* and Broderbund's *Bank Street Writer.* This compatibility is accomplished via a text utility program which also comes on the *PC III* disks. (Previous conversion steps may be needed before running the text utility routine, depending on which type of file is being prepared.)

The program also features mail merge without an additional database—form letters are created and variable data fields passed through them. The process is clean, not requiring an inordinate amount of time. The results are first rate.

Besides being able to add new words to the spell checking dictionary, users can create an auto expander/corrector list. This feature scans a document looking for key words identified in the user-defined list. When it finds the "trigger," it substitutes the full (expanded) phrase for the key word. For instance, finding "PC" in the text could trigger the program to replace "PC" with "PaperClip." This list, like the dictionary, can be directly edited. There are even more bells and whistles to the spell checking subroutine.

A complete telecommunications module is included in *PC III*. With it, modem mavens can autodial online services, download files, and edit them, without ever leaving the program. All RS232 parameters are set via pull-down menu options—a great convenience. Modems supported include the Vicmodem 1600, the 1650, Commodore's 300/1660, Mighty Mo, HesModem II, Hayes and compatibles, and Commodore's 1200/1670. A general RS232 option is available for hardwired schemes.

The more technically minded users will be glad to know that they can choose between several file transfer protocols, XON/XOFF, XMODEM CRC, XMODEM, or Punter C1. As you can see, this subroutine is so comprehensive it alone is worth the $49.95 list price for the entire package.

There's more: A basic outliner makes "thought processing" a reality. Several levels of offset (indentation) are available, all dictated by the style chosen. For example, a business proposal or term paper could be outlined first, then "fleshed-in." The first level might call for uppercase Roman numerals (style), the second for uppercase alpha characters, and a third for lowercase alpha characters. Such an outline would look like this:

SEELEY DAM PROJECT 1A

I. Altering the flow of White River
  A. Digging dry channels
  B. Constructing temporary flow gates
    a. Driving steel caissons
    b. Securing air pressurization equipment

Unlike the telecommunications module, the outliner is not a full-fledged subroutine. Moves, copies, deletes, etc., can be accomplished with the regular text editing commands; however, there is no facility to fold or reopen levels. Several other features found in dedicated outline programs are not available. Still, it is very useful, and provided at no extra cost.

The last group of features includes underlining, super- and subscripting, boldfacing, italicizing, automatic table of contents creation (great!), very flexible headers and footers, centering, multiple justification modes, multiple line spacing, chapter numbering, and automatic page numbering. Whew!

1, 2, 3: Edit, Preview, Print

PaperClip III is not a What-You-See-Is-What-You-Get (WYSIWYG) word processor. During the text entry and edit process, all format commands display onscreen, their effects do not. After issuing the commands to underline a phrase, an uppercase U appears in reverse video image before the phrase and a lowercase u appears after. While this mnemonic coding helps you visualize the layout, it is a poor substitute for seeing the results.

To that end a video preview mode is available via menu option or CTRL command. 40 column monitors will preview in 40, 80, or 160 column modes; RGB and specially equipped composite monitors display 80, 160, and 320 columns. The 160 and 320 column modes do not actually display text, they merely show reverse video image blocks which match the text locations. It is also possible to make certain monitors display in an interlace mode, that is, 50 lines instead of the typical 25. This causes some flicker, but the text is legible.

The results of all format, layout, and related commands appear in the video preview mode, include the colors assigned to boldface, outline, italicized text, etc. The preview scroll is fast and accurate; it can be paused and restarted at will. Printouts will mimic the video preview mode as long as your printer is capable of handling the commands

issued.

Run the Presses

PaperClip III is a solid piece of work, providing a good number of professional level tools at an unbelievable price. The menus are a welcome addition, insuring that the program will stay at the forefront of 64/128 technology.

Batteries Included/Electronic Arts, 1820 Gateway Drive, San Mateo, CA 94404 (phone: 415-571-7171).

–Ted Salamone

geoPUBLISH
Berkeley Softworks
Commodore 64
Disk; $69.95

The industry pundits said it couldn't be done – a real desktop publishing (DTP) program for the C-64. Since developers have long been dispelling myths about the 64's capabilities, it was only natural that someone would bring the hottest rage to the best-selling family of micros. That the program would be so well done is the real surprise.

Besides the CPU, modern day Mergenthalers need *GEOS* 1.2 or higher, a 1541 or 1571 disk drive, and a mouse or a joystick. Naturally a printer is required; after all, that is the program's raison d'être. The extra memory in a RAM expansion unit is used to good advantage, allowing more data and increasing processing speeds.

Postscript, Printers & Plans

Probably the biggest surprise is that *geoPublish (gP)* not only works with all standard GEOS printers, but also supports the *PostScript*-driven Apple LaserWriter!

By supporting the printer/print language that set the desktop publishing standard, *gP* takes a big step into the major leagues. It makes the entire effort more credible, even if most 64/128 owners don't or never will own a LaserWriter.

The ownership issue isn't even important when you consider that numerous print shops own LaserWriters and sell output for 50 cents to $2.00 a page. Some even have trained staff members who can help with the quality of the output.

While this appears to be a good alternative to an expensive investment, there are two flies in the ointment. First, most of these shops use Macintoshes, followed in popularity by IBM PCs. Therefore, on the surface, one would consider this tack a dead end.

Not necessarily so. For the price of an interface cable and a little negotiating, Commodore owners (try this in user group strength for better results) should be able to make suitable arrangements. After all, money has a way of smoothing out the snags.

The other problem concerns getting a 64 or 128 to the print shop. A portable SX-64 comes to mind – it would make the job a whole lot easier. Dig them out of the closets or haunt the flea markets. SX's are more valuable than you think.

Considering the time spent and the relatively small investment ($400 to $500 for an SX-64 and the cable versus $3000 to $4000 for the LaserWriter), the course of action outlined above makes good sense for individuals as well as user groups.

Back to the Show

While not the real equivalent of *PageMaker* on the Mac or *Professional Page* on the Amiga, *geoPublish* packs quite a punch. (The other machines are running with 15 or more times the memory and have floppy drive capacities far exceeding those of a 1541 or a 1571.)

gP sports an internal text editor, provides a powerful, object-oriented draw routine, directly accepts all *GEOS* word processing files, and accepts other Commodore word processing files after conversion with the included Text Grabber.

Like the (much) more expensive DTPs, *geoPublish* provides snap-to grids and user-definable guidelines, mixes text and graphics, supports graphic sizing, cropping, and alteration, and allows for the automatic flow of text around graphics.

Multiple, user-defined columns are a reality. Pages can be viewed in their entirety or zoomed in on; an X/Y window indicates cursor location in extreme detail. The latter display makes close-fitting placement of objects a dream instead of a nightmare. In case the mouse or joystick isn't sensitive enough, the cursor keys can be called into play. Of course the snap-to option makes sure text and graphic objects are properly placed.

Numerous fonts and type sizes are supported. A library of predefined "masters" is provided; non-*GEOS* graphics can be used if you have DeskPack with Graphics Grabber. Drawings can be made opaque or transparent; headlines can be up to 2½" high; automatic page numbering and issue dating are supported.

The list of features is longer still, but you get the picture by now.

Go with the Flow

By setting up "master" pages (a.k.a. templates or style sheets), users can store predefined layouts for later recall. These "masters" make certain the same format is followed for each issue of a publication, insuring design integrity and consistency. Once recalled, text and graphics can be added, the end result being saved to a document file separate from the "master" file.

The actual layout is done by switching between the Page Layout and the Page Graphic modes. Text is loaded and edited here, graphics altered and added. Each mode has its own set of

pull-down menu options and icon toolboxes.

Switching between the modes can be confusing until you learn what is done in each. Like most things in life it's a matter of establishing a routine and sticking to it.

The tutorial brings users through the creation of a two page newsletter called "The Jelly Roll," the inhouse organ for a mythical bakery. All the basics, and a good number of the intermediate and advanced functions are covered. To speed the learning process along, the tutorial makes use of sample text and graphic files. Everything runs smoothly, the flow of events (to build a document) becoming evident once the entire exercise is completed.

The Manual, etc.

Before the tutorial, instructions are given on how to back up the program disk and how to make working copies. Following the procedures, and using the *GEOS* disk copy utility, required 11 disk swaps—for each side of the *gP* disk! You see, extra fonts are stored on the backside.

After the tutorial comes a more in-depth section on using the program. Topics touched on during the tutorial are explained more fully, screen diagrams are more specific, and the relationship between the different modes is discussed.

The Reference Section covers file handling techniques, keyboard short cuts, functions by menu, and working with the graphic support tools, and provides a list of files on the disk. It also explains how to set the automatic date and time parameters and illustrates the library of "master" pages.

A terms dictionary provides insight into new words and phrases, an error message listing does the same for problem identification, and an adequate index makes it easy to find features, functions, and topics.

Respect—Performance

The only unusual thing I noticed about *gP* is that it automatically updates early *geoWrite* files to version 2.1 standards. If you don't have that version of the word processor you can't reaccess the files, unless you save them twice and use one set for word processing and the other for DTP. This oddity was not a surprise, however; the manual and the marketing literature clearly state the ramifications of the automatic update. The manual even suggests saving the files twice if you don't want to upgrade.

Using *geoPublish* on my 64 felt a little strange after working with DTPs on my 2.5 meg Amiga. To say that I doubted its usefulness would be an accurate statement.

But I recognize a good thing when I see it. Since the software behaved in exemplary fashion, did everything promised of it, and produced a good document, it earned my respect. Performance has that effect on me.

Berkeley Softworks, 2150 Shattuck Avenue, Berkeley, CA 94704 (phone: 415-644-0883). *—Ted Salamone*

LANDMARK V2.0
P.A.V.Y. Software
Commodore 64
23 disks; $164.95

Landmark is a disk-based Bible with concordance that allows you to read, underline, and annotate the entire King James Bible for your personal use. It has the ability to load and save files that you've worked on, in order to keep a personal Bible that evolves with you as you read it.

It has a number of useful features: commands like HUNT (short search through current text), REFER (check references in separate text area), ERASE, HOME, and COPY, as well as a host of others. Some commands are useful, others are frills.

The complete King James Bible occupies 23 one-sided disks. I realize that it required a lot of typing to transfer all that Biblical text to disk, but there are many software packages selling currently for $40 that necessitated a greater deal of development and research in order to bring to market.

Although the editor is described in the documentation as a word processor, it is actually a sophisticated text editor, lacking many features that are mandatory on more advanced products. It is on the slow side, and requires patience. The documentation indicates that *Landmark* is 100% machine language, but it seemed to be very sluggish on some functions, as if it were a compiled program. The text area is little more than a few pages long, and it rapidly becomes annoying to wait through the many loads that are needed to examine even a single chapter.

I have been a student of the Bible for over eight years. I have never felt the need to read it from a monitor screen—it seems to depersonalize the meaningful content therein. But I realize that there may be a demand for this type of software.

If I did go shopping, I would expect much, much more from a package of this expense. One sorely missed feature is a speed-loader DOS that would decrease access time and allow the user to spend less time staring at the LOADING prompt.

If you are a student of the Bible, or a theology professor who must have a software-based Bible for your 64, it would appear that *Landmark* is a necessary purchase, considering that it is the only package of its kind.

Landmark does include a handsome plastic container with vinyl leaves to hold the entire disk set. Versions for the 128 and 64 are on the same disk.

Although this product was satisfactorily complete, you will find that its purchase is ultimately a personal choice, based on your own needs and your financial resources.

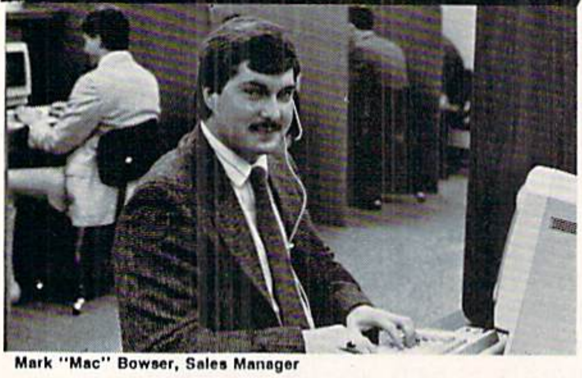
P.A.V.Y. Software, P.O. Box 1584, Ballwin, MO 63022 (phone: 314-527-4505). *—Cleveland M. Blakemore*

Lyco Computer

Marketing & Consultants

Since 1981

Lyco Means Total Service.



Mark "Mac" Bowser, Sales Manager

I would personally like to thank all of our past customers for helping to make Lyco Computer one of the largest mail order companies and a leader in the industry. Also, I would like to extend my personal invitation to all computer enthusiasts who have not experienced the services that we provide. Please call our trained sales staff at our toll-free number to inquire about our diverse product line and weekly specials.

First and foremost our philosophy is to keep abreast of the changing market so that we can provide you with not only factory-fresh merchandise but also the newest models offered by the manufacturers at the absolute best possible prices. We offer the widest selection of computer hardware, software and accessories.

Feel free to call Lyco if you want to know more about a particular item. I can't stress enough that our toll-free number is not just for orders. Many companies have a toll-free number for ordering, but if you just want to ask a question about a product, you have to make a toll call. Not at Lyco. Our trained sales staff is knowledgeable about all the products we stock and is happy to answer any questions you may have. We will do our best to make sure that the product you select will fit your application. We also have Saturday hours — one more reason to call us for all your computer needs.

Once you've placed your order with Lyco, we don't forget about you. Our friendly, professional customer service representatives will find answers to your questions about the status of an order, warranties, product availability, or prices.

Lyco Computer stocks a multimillion dollar inventory of factory-fresh merchandise. Chances are we have exactly what you want right in our warehouse. And that means you'll get it fast. In fact, orders are normally shipped within 24 hours. Free shipping on prepaid orders over $50, and there is no deposit required on C.O.D. orders. Air freight or UPS Blue/Red Label shipping is available, too. And all products carry the full manufacturers' warranties.

I can't see why anyone would shop anywhere else. Selection from our huge in-stock inventory, best price, service that can't be beat—we've got it all here at Lyco Computer.

TO ORDER, CALL TOLL-FREE: 1-800-233-8760
New PA Wats: 1-800-233-8760
Outside Continental US Call: **1-717-494-1030**

Hours: 9AM to 8PM, Mon. - Thurs.
9AM to 6PM, Friday — 10AM to 6PM, Saturday

For Customer Service, call 1-717-494-1670,
9AM to 5PM, Mon. - Fri.
Or write: Lyco Computer, Inc.
P.O. Box 5088, Jersey Shore, PA 17740

VISA MasterCard

C.O.D. Risk-Free Policy: • full manufacturers' warranties • no sales tax outside PA • prices show 4% cash discount; add 4% for credit cards • APO, FPO, international: add $5 plus 3% for priority • 4-week clearance on personal checks • we check for credit card theft • sorry, compatibility not guaranteed • return authorization required • due to new product guarantee, return restrictions apply • price/availability subject to change • prepaid orders under $50 in Continental US, add $3.00

Monitors

Thomson:

230 Amber TTL/12"	$79.95
4120 CGA	$225.95
4160 CGA	$259.95
4460 EGA	$319.95
4375 UltraScan	$389.95
GB 100 EGA Card	$129.95
GB 200 Super Card	$219.95

Blue Chip:

BCM 12" Green TTL	$64.95
BCM 12" Amber TTL	$69.95

Magnavox:

BM7652	$84.95
BM7622	$84.95
7BM-613	$79.95
7BM-623	$79.95
CM8502	$189.95
CM8505	$209.95
CM8562	$239.95
CM8762	$249.95
8CM-515	$269.95

NEC:

Multisync II	$599.95

Save $210 over NEC Multisync with Thomson 4375 UltraScan $389.95

Modems

Avatex:

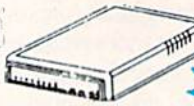
1200e	$69.95
1200i PC Card	$69.95
1200hc Modem	$89.95
2400	$179.95
2400i PC Card	$169.95

Hayes:

Smartmodem 300	$149.95
Smartmodem 1200	$285.95
Smartmodem 2400	$425.95

Smarteam

1200 Baud Modem



OUR PRICE $89.95

Hayes Compatible

COMMODORE

Access:

Echelon	$25.95
Mach 5	$19.95
Mach – 128	$28.95
10th Frame	$22.95
Triple Pack	$11.95
Wld. Cl. Leader Brd.	$22.95
Famous Courses #1	$11.95
Famous Courses #2	$11.95
Leader Board Pack	$14.95

Action Soft:

Up Periscope	$18.95
Thunderchopper	$18.95

Activision:

Champion. Basketball	$19.95
Music Studio	$19.95
Leather Goddesses	$22.95
Top Fuel Eliminator	$15.95
Beyond Zork	$25.95
GFL Football	$19.95
Gee Bee Air Rally	$16.95
Last Ninja	$19.95
Might & Majic	$19.95
Nord & Bert	$19.95

Batteries Included:

Paperclip III	$31.95
Outrageous Pages	$31.95

Berkeley Softworks:

Geofile C64	$29.95
Geo Calc C64	$29.95
Geos 64	$35.95
Geos 128	$39.95
Geowrite	$29.95
Geowrite 128	$39.95
Geopublish C64	$39.95
Berkeley TriPak	$29.95

Broderbund:

Bank St. Writer	$27.95
Carmen San Diego	$19.95
Graphic Lib. I, II, III	$13.95
Print Shop	$25.49
Print Shop Compan.	$22.95
Print Shop Paper	$12.95
Cauldron	$16.95
Superbike Challenge	$11.95

Strategic Simulations:

Gettysburg	$33.95
Phantasie II	$22.95
Phantasie III	$22.95
Ring of Zelfin	$22.95
Road War 2000	$22.95
Shard of Spring	$22.95
Wizards Crown	$22.95
Wargame Constr.	$16.95
Battlecruiser	$33.95
Battle of Antetiem	$28.95

COMMODORE

Electronic Arts:

Bard's Tale	$25.95
Legacy of The Ancients	$20.95
Marble Madness	$20.95
Starfleet I	$25.95
Yeager's AFT	$22.95

Epyx:

Destroyer	$22.95
Fastload	$19.95
Sub Battle	$22.95
Winter Games	$11.95
California Games	$22.95
Graphics Scrapbook	$9.95
Str. Sports Basketball	$22.95
Summer Games II	$11.95
World Games	$22.95
Jet Combat	$8.95

Firebird:

Elite	$18.95
Golden Path	$22.95

AMIGA

Access:

Leader Board	$22.95
Tournament #1	$11.95

Activision:

Champ. Basketball	$25.95
Championship Golf	$22.95
GFL Football	$25.95
Music Studio	$27.95

Electronic Arts:

Deluxe Print II	$48.95
Rockford	$25.95
Empire	$31.95
Scrabble	$25.95

Epyx:

Apshai Trilogy	$11.95
Rogue	$22.95
Winter Games	$22.95
World Games	$22.95

Firebird:

Guild of Thieves	$25.95
Pawn	$25.95
Starglider	$25.95

Microprose:

Silent Service	$22.95

Sublogic:

Flight Simulator II	$31.49
Jet Simulator	$31.49
Scenery Disk	$CALL

Unison World:

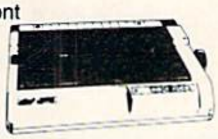
Print Master	$19.95
Art Gallery 1 or 2	$14.95
Fonts & Borders	$17.95

1-800-233-8760

New for '88 — Commodore Ready — Great Price

star micronics·inc NX-1000

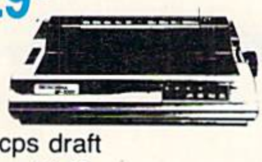
- 144 cps Draft
- 36 cps NLQ
- EZ Operation Front Panel Control



$179.95

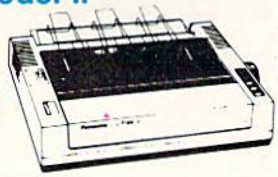
SEIKOSHA SP 180VC

$129.95



- 100 cps draft
- 20 cps NLQ

Panasonic 1091 Model II



$199.95

- 192 cps Draft
- 32 cps NLQ

PRINTERS

star micronics·inc

NX-1000	$179.95
NX-1000C	$179.95
NX-1000 Color	$225.95
NX-1000C Color	$229.95
NX-15	$309.95
NR-10	$339.95
NR-15	$439.95
NB-15 24 Pin	$699.95
NB24-10 24 Pin	$425.95
NB24-15 24 Pin	$579.95

CITIZEN

120 D	$169.95
180 D	$189.95
MSP-10	$259.95
MSP-40	$309.95
MSP-15	$349.95
MSP-50	$399.95
MSP-45	$459.95
MSP-55	$539.95
Premiere 35	$499.95
Tribute 224	$649.95

SEIKOSHA

SP 180Ai	$129.95
SP 180VC	$129.95
SP 1000VC	$139.95
SP 1000AP	$169.95
SP 1200VC	$155.95
SP 1200Ai	$165.95
SP 1200AS RS232	$165.95
SL 80Ai	$299.95
MP1300Ai	$269.95
MP5300Ai	$399.95
MP5420Ai	$879.95
SP Series Ribbon	$7.95
SK3000 Ai	$339.95
SK3005 Ai	$419.95
SPB 10	$CALL
SL 130Ai	$599.95

Toshiba

321SL	$489
341 SL	$659
P351 Model II	$899
351 SX 400 cps	$1019

EPSON

LX800	$179.95
FX86E	$279.95
FX286E	$424.95
EX800	$399.95
LQ500	$309.95
LQ1000 w/Tractor	$549.95
LQ2500	$819.95
GQ3500	$LOW
LQ850	$489.95
LQ1050	$659.95

DIABLO

D25	$499.95
635	$779.95

BROTHER

M1109	$195
M1409	$299
M1509	$335
M1709	$475
Twinwriter 6 Dot & Daisy	$899
M1724L	$599
HR20	$339
HR40	$569
HR60	$709.95

Panasonic

1080i Model II	$179.95
1091i Model II	$199.95
1092i	$319.95
1592	$409.95
1595	$459.95
3131	$299.95
3151	$479.95
KXP 4450 Laser	$CALL
1524 24 Pin	$559.95
Fax Partner	$589.95

OKIDATA

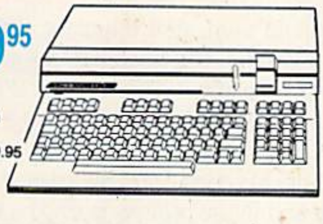
an OKI AMERICA company

Okimate 20	$119
Okimate 20 w/cart	$179.95
120	$189.95
180	$219.95
182	$209.95
182 +	$225.95
183	$249.95
192 +	$309.95
193 +	$449.95
292 w/interface	$449.95
293 w/interface	$585.95
294 w/interface	$819.95
393	$955.95

COMMODORE 128 D System

$449.95

Commodore 128D Plus Thomson 4120 Monitor $649.95



COMMODORE HARDWARE

1541 II Disk Drive	$175.95
1581 Disk Drive	$189.95
Indus GT C-64 Drive	$169.95
1802C Monitor	$189.95
64 C Computer	$169.95
128D Computer/Drive	$449.95
C-1351 Mouse	$32.95
C-1750 RAM	$CALL
1764 RAM C64	$117.95
1084 Monitor	$279.95
128 Computer	$219.95

LASER COMPACT XT

- PC-XT Compatible
- 4.77 - 8.00 Mhz Super Turbo Clock Speed
- Built-in 5¼ Drive
- Built-in RGB Video Output
- Parallel Printer Port
- Serial RS232
- Joystick/Game Port

$475.95



COMMODORE 128 EXCEL 2001 Special

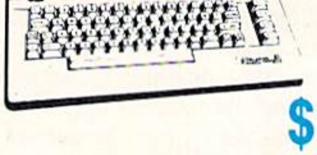


$409.95

Seagate HARDWARE

ST225 20 meg hard drive $215.95★
ST251 40 meg hard drive $399.95★
★DTC Controller Kit for PC/XT $39.95
ST238 30 meg hard drive .... $249.95★★
★★DTC RLL Controller Kit for PC/XT $49.95
Add $10.00 for Western Digital Controllers
ST-125 20 meg 3.5 hard drive $289.95
DTC Controller Kit for PC/XT $39.95
ST-125 20 meg internal card w/controller $349.95

COMMODORE 64C Computer



$169

Attention Educational Institutions:
If you are not currently using our educational service program, please call our representatives for details.

Reader Service No. 145

COMAL COLUMN

I PROGRAM, THEREFORE I AM

By Richard Herring

Well, I've been asking you for letters and you've obliged. In future columns, we will all share in some of your discoveries. This month, though, I'll try to answer the most persistent question posed in the letters I've received from readers across North America (and beyond).

Starting with Edward Leduc in Canada, then Robert Neer in New York, and continuing south over the summer, the theme of that question has been, "How do you learn to program a computer?" Variations have included "What's the best way to learn COMAL?" and "Why should I learn COMAL rather than some other language?"

Without trying to psychoanalyze why we all bought our computers in the first place, I think I can describe the four most common groups that people learning to program fall into. There are typists, students, hackers, and consultants.

The typist doesn't know how he learned to program. He subscribed to a magazine after he bought his computer and figured out that those program listings in the back were often as good as the stuff down at the SoftWare-R-Us Emporium. So he started typing.

He began to learn syntax through osmosis. The keywords became obvious. The style and technique of good programmers began to rub off. Pretty soon, he's thinking "This is okay, but if I just tweek it here, add a subroutine there, it'll do just what I want." So he does, it does, and a programmer is born.

For the typist, COMAL may be a difficult language to learn because program listings in books and magazines are dominated by BASIC. Yet COMAL provides an ideal environment for the typist because it is such a logical language. (How do you end a FOR structure? With an ENDFOR command.) It encourages procedural programming and a one-command-per-line approach so it is much easier to decipher than BASIC.

The student is the mirror image of the typist. The student starts out wanting to learn to program and does so in an organized, orderly fashion. Whether he's in or out of school, he goes through the tutorial a chapter at a time. Some sections don't make much sense or don't seem to have any real use, but he diligently learns them anyway.

Throughout the process, lights go on for the student, concepts fit together, and the individual pieces come to comprise the whole. In the end, the student may or may not be a better programmer than the typist (who learned by example from many good programmers rather than the one who wrote the book). But the student has a balanced overview of math functions, string handling, I/O, sprites, all those things.

For the student, several of the books we reviewed in the October '87 *COMAL Column* can fit the bill. Try the *COMAL Workbook* by Gordon Shigley, *Foundations in Computer Studies With COMAL* by John Kelly, *Starting with COMAL* by Ingvar Gratte, *Beginning COMAL* by Borge Christensen, or the *Cartridge Tutorial Binder* (for version 2.0).

COMAL is especially good for the student because it forces structured programming and also because it provides a good base for a second language like Pascal, BASIC, or LOGO.

Next we come to the hacker. Were it not for software copy protection schemes, this method of learning to program would take a real dive. The hacker wants to copy software. He may never play that ex-copy-protected game, but by golly, he has it on a disk with four other broken games.

As these defeated protection schemes mount up, the hacker is learning like a madman. Disk formats. I/O. A bit of assembly language. File types.

The friendly neighborhood hacker owns all the copy programs (especially if they *were* protected) but uses none of them. No, his disk full of kludgy programs works better and is more comfortable, more powerful, and more fun (often with a bit of fanfare as his name scrolls across the screen).

Of my two closest ex-hacker friends, one turned into the sysop of a really good bulletin board (that *never* allows bootleg software); the other is a full-time programmer whose AI work is fascinating.

The consultant, unpaid and often reluctant, isn't really a programmer—never will be. But he knows programming well enough to answer questions. The consultant bought his computer to do a specific task...or maybe just because his yup friends had one.

The consultant got into his computer. He learned to make canned programs sing and in the process learned a bit about hardware and a smidgen about programming. As the family/neighborhood/school/office expert, he learned to love answering questions. "Push this, pull that, and voilà." He feels so good as he hands you your 1986 taxes that you couldn't get out of the #$% computer or as he watches your modem work for the first time.

He hates to disappoint. Eventually it happens. A question on real programming. A spouse/neighbor/student/coworker needs to sort a file, create a screen display, or write a little calculator program. So, avoiding the dreaded "Huh?" at all cost, he says "I'll get back to you." Next day he has an answer to go with his red-rimmed eyes and his ad hoc programming career has begun.

COMAL is great for the consultant because COMAL

Tech (sort of) Notes

1) Including comments in your program code is considered good programming practice. (By everyone except paid consultants. They need to make sure they are the only ones who can work on their programs.) Programming languages may be rated as:

–Poor, if comments must be on separate lines from the code they document.

–Fair, if comments can only be included at specific points within statements.

–Good, if comments can be placed to the immediate right of the statements they document.

–Excellent, if comments are delimited by a single character.

In COMAL, comments may be placed on a line by themselves or at the end of any program line. Comments always begin with a double slash (//); however as you are programming you can type a single exclamation point (!) and COMAL will convert it. Comments in COMAL do not slow down your program, except to make it load a bit slower since it will be longer. I guess COMAL rates an excellent for its commenting capabilities. The only thing you can not do is imbed a comment in the middle of a program line so that program code both follows and precedes it.

2) Why do people fail to learn to program? Key reasons seem to be:

–Some fail to understand how stupid and how literal computers are. They see the syntax of the language, but don't follow it, thinking that the computer should understand. They are liable to substitute "INTEGER" for the command "INT" and expect the program to run.

–Some have no reason to learn. Much as most of us would like to be fluent in French or Spanish or some other foreign language, few of us will devote the time. The same is true for computer languages.

–Fear. What if my program blows up the computer? Or changes my bank balance? Some people get it in their heads that learning to program is different from learning anything else. It isn't. Like everything else, it is painful when you are memorizing and elating when you're making things happen.

3) Although a structured program has no need for GOTOs or for labels, writing a program without GOTOs does not make it structured. Some tests of a structured program are:

–Does each program module represent a complete thought of no more than one printed page for the sake of readability?

–Does the program listing use indented structures?

–Are comments used for documenting procedures and variables?

–Did the programmer prefer readable code to maximum efficiency?

–Was the program developed top down, beginning with a logical description in English and ending with a program in the computer's language?

–Does the program use only the control structures designed for the language?

makes non-trivial tasks easy and fast. A few lines of code can accomplish a lot. And a procedure, once written, is good in the next program. Not until Commodore's upgrade to BASIC 7.0 did it add most of the additional commands available in COMAL. Even at that, BASIC is plagued by its history of limited identifiers and runtime slowness.

So that's how people, at least the people I know, learn to program. For all of them, except possibly the hacker who may go first to assembly language, COMAL provides a perfect learning environment.

Remember when you got your computer and entered your first program from the owner's manual? After 10 or 15 minutes of typing and debugging, you were able to make the computer go "bzzzlattt." After that experience, writing a major program seemed as possible as finding a video arcade that takes nickels.

COMAL creates the possibility. For the typist, it makes program modification easy. The student can revel in structure. The consultant will generate answers faster and more efficiently. If someone will only write some good COMAL copy protection schemes, even the hacker will be able to play.

For those of you who have begun to program in BASIC and are wondering whether to try COMAL, let me paraphrase RF Tolli, the sysop of the local Commodore BBS. "I suspect the reason they all stick to BASIC is to have something to complain about."

That's all swell. Now you can analyze your friends. But, how should you learn? Without knowing you personally, I'd suggest the following casual approach.

1) Read carefully the manual that came with your computer as it relates to everything *except* programming. Learn all about formatting disks, clearing memory, etc. Become as familiar with your computer as possible.

2) Go to meetings of a local user's group if there is one. Talk to people about what they're doing. Get a feel for what kinds of things the beginners are into and what kinds of tasks only the more experienced programmers will tackle.

3) Type in any program listings you find. Don't worry about what you don't understand. The point is that you'll be getting a feel for the syntax of the language, for the construction of a good program, and for the commands that the language uses.

4) From the user's group, or from a mail order outfit, get five or ten disks of public domain software written in the language you're learning. Play with it. List out the programs to see how they're built.

5) When you're comfortable, try modifying a program you have typed in or a public domain program. See what happens. If your change doesn't work, drag out the manual and try to find out why. Through the process, focus more on making it work than on understanding why it works. Once you've made it happen a few times, you will realize that somewhere in the process you came to an intuitive feeling about why the program produces those particular results.

6) Set yourself a small programming goal. Something you think can be done in fewer than 100 lines. Make it work. Then add bells and whistles to your heart's content.

Next time we will look at another good reason to learn COMAL–its speed. As you explore COMAL, feel free to share your questions and your insights with me (P.O. Box 1544, Tallahassee, FL 32302). ☐

COMMODARES
PROGRAMMING CHALLENGES

By Dale Rupert

Each month, we'll present several challenges designed to stimulate your synapses and toggle the bits in your cerebral random access memory. We invite you to send your solutions to:

Commodares, c/o *Ahoy!*
P.O. Box 723
Bethel, CT 06801

We will print and discuss the cleverest, simplest, shortest, most interesting and/or most unusual solutions. Be sure to identify the *name* and *number* of the problems you are solving. Put *your* name and address on the listings as well. Show sample runs if possible. Briefly describe your solutions and tell what makes them unique or interesting, if they are. You must enclose a stamped, self-addressed envelope if you want any of your materials returned. Solutions received by the middle of the month shown on the magazine cover are most likely to be discussed, but you may send solutions and comments any time. Your original programming problems, suggestions, and ideas are equally welcome. The best ones will become *Commodares*!

PROBLEM #54-1: JOYSTICK INITIALS

This problem was suggested by Jeff Veasey (Lovington, NM). Let the user enter his three initials on the screen by using the joystick as in the arcade games. Pressing the joystick right or left cycles up or down through the alphabet at the position of the cursor on the screen. Pressing the joystick button selects the displayed letter and moves the cursor one space to the right. Make this a no-frills solution. If you are ambitious, include the "back-arrow" character between A and Z which allows the user to move the cursor to the left to correct one of the previously selected initials.

PROBLEM #54-2: FILE LOCATOR

Try this one from Dan Mooney (Hinesville, GA). Write a program to find the starting and ending addresses of a selected disk file. Dan says he has such a program but it takes up to 20 minutes for a long file. He wonders if someone knows how to use burst mode on the 1571 or 1581 drives to speed this up. Burst mode or not, let's see your quickest solutions.

PROBLEM #54-3: WRITTEN REMAINDER

Gene Majewski (Bellwood, IL) suggested this problem. The user enters two whole numbers. The computer gives the integer quotient and the remainder. For example, if the user enters 100,40 the computer responds 100 / 40 = 2 REMAINDER 20.

PROBLEM #54-4: BIRTHDAY PARTY

Here's another good challenge from Necah Buyukdura (Ankara, Turkey). Write a BASIC program to determine who will attend my birthday party, on the basis of these five premises:

1. My friends Bert and Carl cannot tolerate each other. Only one or the other, but not both, will be there.
2. If Adam comes, then Bert will be there also.
3. Either Duke or Emil, or both will come.
4. Duke told me that he would come only if Carl was also coming.
5. If Emil comes to my party, certainly neither Adam nor Duke will fail to come.

This month we will look at some of the more interesting solutions to *Commodares* from the January 1988 issue of *Ahoy!* I mentioned last month that I generally ignore suggestions for problems which we have previously published, since some readers have been solving these problems since the first issue of *Ahoy!* in January of 1984. The first problem we will discuss is an exception. This was a repeat of a *Commodare* in the February 1985 issue of *Ahoy!*

Looking back through my files, I found that there are only two readers who submitted solutions to *Commodares* in the February 1985 issue as well as in this issue three and a half years later. Those two readers deserve special recognition. Congratulations to Jim Speers of Niles, Michigan and to Wallace Leeker of Lemay, Missouri. The rewards for such achievements are the usual for this column: self-satisfaction of a job well done and the honor of seeing your names in genuine print. Thanks for all the solutions and suggestions. (I hope to see solutions from both of you for yet another three and a half years.)

Richard van Frank (Montclair, NJ) suggested *Problem #49-1: Rome Revisited.* The user enters any integer from 1 to 3999, and the computer returns the corresponding Roman numeral. This solution from Jerry Nichols uses the fewest instructions.

```
•1 REM ===================================
•2 REM    COMMODARES PROBLEM #49-1 :
•3 REM      ROME REVISITED
•4 REM    SOLUTION BY
•5 REM      JERRY NICHOLS
•6 REM ===================================
•10 INPUT"NUMBER";N
•20 READ V,R$: IF N=>V THEN N=N-V: PRINT
 R$;
•30 IF N>0 THEN 20
•60 DATA 1000,M,1000,M,1000,M,900,CM,500,
```

```
D,400,CD,100,C,100,C,100,C
•70 DATA 90,XC,50,L,40,XL,10,X,10,X,10,X,
9,IX,5,V,4,IV,1,I,1,I,1,I
```

Jim Speers and Sal Manfredonia (Dover, NJ) took a similar approach of putting combined symbols such as CD, XL, and IX into DATA statements. This greatly simplifies the program logic, as Jerry's program above shows.

Michael Stype (Michigan City, IN) suggested modifying the program to let the computer count in Roman numerals. You can change line 10 and add line 40 as follows:

```
10 FOR K=1 TO 3999 : RESTORE : N=K
40 PRINT TAB(15); "="; K : NEXT
```

Do you see why we must use the second variable K rather than use an N FOR-NEXT loop?

Paul Underwood (Bridgewater, CT) was kind enough to send Nero's Notes to explain his solution to *Rome Revisited.* (Evidently Nero put his fiddle aside to work on the computer for a while.) Paul mentioned that the Romans used overscores to represent numbers larger than 3999, and that they did not use the numeral 0.

By the way, the use of 0 in a numbering system was quite a revolutionary idea. An encyclopedia points out that the significance of 0 was not understood until the 6th century A.D., and for that reason the Julian calendar established in 46 B.C. has no year 0. In fact Western civilization did not accept 0 until after the work of Fibonacci in 1228.

Returning to the present, let's look at *Problem #49-2: Score Keeper,* submitted by Joseph Sweely (Pennsauken, NJ). This program allows the user to enter a score and his initials. The program saves and displays the top five scores in order along with the initials, as is done with arcade games.

Here is the solution from Craig Ewert (Crystal Lake, IL), another longtime *Commodares* solver.

```
•1 REM ===================================
•2 REM    COMMODARES PROBLEM #49-2 :
•3 REM      SCORE KEEPER
•4 REM    SOLUTION BY
•5 REM      CRAIG EWERT
•6 REM ===================================
•11 GOTO 20
•12 FOR K=0 TO 4: IF SC>HS(K) THEN 14
•13 NEXT: RETURN
•14 INPUT"INITIALS";IN$
•15 FOR I=5 TO K+1 STEP -1:HS(I)=HS(I-1):
HI$(I)=HI$(I-1):NEXT:HI$(K)=IN$:HS(K)=SC
•16 PRINT"[CLEAR]": FOR I=0 TO 4:PRINT I+
1,HS(I),HI$(I): NEXT: RETURN
•20 INPUT"SCORE";SC: GOSUB 12: GOTO 20: E
ND
```

Craig uses the arrays HS (high score) and HI$ (high score initials) to store the data. Line 11 would typically be the body of the game program. Line 12 steps through the current high scores to find the index of the proper position for the new score. The routine at line 15 inserts the new score and initials into the arrays.

Jeff Veasey (Lovington, NM) sent a deluxe solution to this problem which lets the user cycle through the letters to pick his initials with the joystick. That's the basis of the simplified *Problem #54-1* above.

Joseph Sweely also suggested *Problem #49-3: Sprite Flipper.* The problem is to flip over a sprite, that is, reverse it from left to right. The following program is a combination of two solutions sent by Rick Dollar (Bentonville, AR).

```
•1 REM ===================================
•2 REM    COMMODARES PROBLEM #49-3 :
•3 REM      SPRITE FLIPPER
•4 REM    SOLUTION BY
•5 REM      RICK DOLLAR
•6 REM ======  FOR C-64 ONLY ============
•10 PRINT CHR$(147):B1=12288:V=53248:POKE
251,0:POKE252,48:POKE253,64:POKE254,48
•20 DATA 0,0,0,2,255,248,6,224,248,14,211
,249,30,224,250,62,255,255,126,217,250
•30 DATA 254,214,251,126,230,250,62,255,2
55,30,193,250,14,254,249,158,193,248,98
•40 DATA 255,248,64,0,0,128,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0
•50 FOR N=12352 TO 12415:POKE N,0:NEXT
•60 FOR A=B1 TO B1+63:READ B:POKE A,B:NEX
T:POKE2040,192:POKE2041,193:POKEV+39,1
•70 POKE V+40,1:POKE V,160:POKE V+1,100:P
OKE V+21,3:POKE V+2,100:POKE V+3,100
•80 FOR A=828 TO 877:READ B:POKE A,B:NEXT
•90 DATA 160,60,132,2,177,251,32,97,3,200
,200,145,253,136,177,251,32
•100 DATA 97,3,145,253,200,177,251,32,97,
3,164,2,145,253,136,136,136
•110 DATA 16,222,96,162,8,10,110,251,3,20
2,208,249,173,251,3,96
•120 INPUT"PRESS ENTER KEY FOR BASIC VERS
ION";K$
•130 FOR A=0 TO 60 STEP 3:FOR C=0 TO 2:D=
2-C:N=PEEK(B1+A+C):M=0
•140 B2=B1+64: FOR B=7 TO 0 STEP -1:P=2[U
PARROW]B
•150 M=M+(ABS(P=(N AND P)))*2[UPARROW](7-
B):NEXT:POKE B2+A+D,M:NEXT:NEXT
•160 INPUT"PRESS ENTER KEY TO ERASE SPRIT
E 2";K$
•170 FOR N=12352 TO 12415:POKE N,0:NEXT
•180 INPUT"PRESS ENTER KEY FOR M.L. VERSI
ON";K$
•190 SYS 828
```

This program for the C-64 shows the power of machine language programming. It performs the sprite swapping task first in BASIC, then in machine language. When you run the program, the sprite is created and displayed. You are prompted to press the RETURN key to reverse the sprite using the BASIC routine.

Next press the RETURN key to erase the reversed sprite.

Watch closely after pressing RETURN for the third prompt. In less than the blink of an eye, the machine language routine essentially duplicates the process performed by the BASIC routine. There is no noticeable delay from the time you press RETURN until the reversed sprite appears. Press RUN STOP and RESTORE to remove the sprites from the screen.

Line 10 sets up some variables and page zero addresses. The sprite definition data is in lines 20 through 40. Line 50 blanks out sprite number 2, which will receive the reversed data. Lines 60 and 70 perform the sprite initialization and display the first sprite. Line 80 puts the machine language routine in lines 90 through 110 into memory. The BASIC swapping routine is in lines 130 and 140. Line 170 erases the reversed sprite. Line 190 calls the machine language routine stored in the cassette buffer at address 828.

With a monitor program, you can disassemble the machine language routine in addresses 828 through 877 (33C through 36D hex). The routine is not relocatable, so you would have to rewrite parts of it if you wanted to store it somewhere else, such as in the C-128's cassette buffer. You would also need to modify some of the sprite parameters to use this on the C-128.

Jim Speers sent a BASIC program which causes the sprite to flip as quickly as the machine language version. He performed the swapping routine without showing the second sprite. Then at the press of a key, the reversed sprite is instantly displayed. To see this process, add these lines to the program above:

```
200 INPUT K$:POKE 2040,193:POKE 2041,192
210 INPUT K$:POKE 2040,192:POKE 2041,193
220 GOTO 200
```

The secret is simply to exchange sprite data pointers. Address 2040 stores the block number of sprite one data, and similarly address 2041 points to sprite two data. Each press of the RETURN key instantly swaps the areas of memory used to define the two sprites, thereby causing the normal and reversed images to trade places. This little demo should give you some animation ideas.

Jim Borden (Carlisle, PA), another longtime *Commodares* expert, and Frank Colaricci (Winter Park, FL) also sent machine language solutions to this problem. Congratulations to you and the others who solved this challenge.

Wrapping it up this month is *Problem #49-4: Easy As...*, which asked for a program to calculate pi as accurately as possible using only +, −, /, *, and exponentiation. PRINT 22/7 was a start, but *Ahoy!* readers went much further.

Among the many series approximations submitted for pi are these:

1) $\arctan(X) = X - X\uparrow 3/3 + X\uparrow 5/5 - X\uparrow 7/7 \ldots$
   $pi/4 = atn(1/2) + atn(1/3)$
2) $\arcsin(X) = X + (X\uparrow 3)/6 + 3^*(X\uparrow 5)/40 + 15^*(X\uparrow 7)/336 \ldots$
   $\arcsin(1/2) = pi/6$
3) $pi = 4/1 - 4/3 + 4/5 - 4/7 + 4/9 - 4/11 \ldots$
4) $pi = 2^*(2/1)^*(2/3)^*(4/3)^*(4/5)^*(6/5)^*(6/7) \ldots$
5) $pi = 4/(1+1^*1/(2+3^*3/(2+5^*5/(2+7^*7/(2+9^*9/(2\ldots))))))$

This last one is a continued fraction and must be evaluated from right to left. Justin Smalley (Boulder, CO), another reliable solver, mentioned that this leads to a very time-consuming calculation (three hours to get 7 decimal places!). Justin said that this continued fraction was described in *Science News*, 4/25/87, in an article about the great Indian mathematician S. Ramanujan.

Justin sent another of Ramanujan's formulas involving complex numbers in which the first term alone gives pi accurate to seven decimal places.

Ellis Hormats (Maitland, FL), one more *Commodares* veteran, mentioned an article in *Scientific American*, February 1988, pp. 112-117, which discusses several methods that have been used to calculate pi to many decimal places (at least 1 million).

Ellis used a Monte Carlo method to find pi. The idea is to randomly pick points within a square in which a circle is inscribed. The ratio of the number of points which fall inside the circle to the total number of points is pi/4, since that is the ratio of the area of the square to the area of the circle. In his program below, line 60 calculates the distance of the randomly chosen point from the center of the circle. Line 80 increments P whenever a point is within the circle of radius 1.

```
·1 REM ====================================
·2 REM    COMMODARES PROBLEM #49-4 :
·3 REM       EASY AS ...
```

ERRATUM

Vee Kloros (April 1988)

Several lines of *Vee Kloros* were printed incorrectly. Load in your version of *Vee Kloros* and type in the lines below. Scratch the old copy off the disk directory and save the new version to disk.

```
·760 PRINT" [GREEN][c *][RVSON] [c *][RED][RVSOFF][s C][s C][
 GREEN][RVSON][sEP] [RVSOFF][sEP][RED][8"[s C]"][GREEN][RVSON
 ] [RED]";                                                    JM
·765 PRINT"[RVSOFF][s C][GREEN][RVSON][sEP][RVSOFF][sEP][RED]
 [16"[s C]"][c W]"                                            JD
·770 PRINT' [s B][GREEN][c *][RVSON] [c *][sEP] [RVSOFF][sEP]
 [RVSON][c H][c O][c O][c H][c O][c O][RVSOFF][3" "][RVSON][S
 S] [sEP][RVSOFF][sEP] [RVSON]";                              HN
·775 PRINT"[c H][RVSOFF]  [RVSON][c H] [c L][c G][RVSOFF][c U
 ][RVSON] [RVSOFF][c J][RVSON][c H] [c L]  [RVSOFF] [RED][s B
 ]"                                                           AO
·780 PRINT" [s B] [GREEN][c *][RVSON][SS] [RVSOFF][sEP] [RVSO
 N][c H][c U][RVSOFF] [RVSON][c H][c U][RVSOFF][4" "][RVSON][
 SS] [RVSOFF][c *][RVSON]";                                   EA
·785 PRINT"[c *][RVSOFF] [RVSON][c H][RVSOFF]  [RVSON][c H][R
 VSOFF] [RVSON][c L][c G][RVSOFF][c Y][c *][RVSON][c *][c H][
 RVSOFF] [RVSON][c L][RVSOFF][c *][RVSON][c *][RVSOFF] [RED][
 s B]"                                                        PL
·790 PRINT" [c Q][s C][s C][GREEN][c *][sEP][RED][s C][s C][G
 REEN][RVSON][c H][c U][c U][c H][c U][c U][RED][RVSOFF][3"[s
  C]"][GREEN][RVSON]  [RED][RVSOFF][s C]";                    AG
·795 PRINT"[GREEN][c *][RVSON][c *][c H]  [c H] [c L][c G][RE
 D][RVSOFF][s C][s C][GREEN][RVSON] [c H] [c L]  [RED][RVSOFF
 ][s C][c W]"                                                 LI
·800 PRINT" [s B][15" "][GREEN][RVSON][c O][c O][RVSOFF]  [c
 *][5" "]";                                                   CO
·805 PRINT"[4" "][RVSON] [RVSOFF][6" "][RED][s B]"            PF
·810 PRINT" [s B][8" "][YELLOW]JOYSTICK IN PORT TWO[8" "][RED
 ][s B]"                                                      DF

·830 PRINT" [s B][WHITE]APPROACHING V-KLS,THE MYSTERY PLANET[
 RED][s B]"                                                   NP

·890 PRINT"[RVSON][GREEN]"T$(T):POKEV+34,C(T*2):POKEV+35,C(T*
 2+1):POKE646,C(T*2+1)+8                                      KD

·930 I$=STR$(HI):SYSPL,32,11:PRINT"[RVSOFF][WHITE]";:GOSUB180
 :RETURN                                                      DP
```

```
•4 REM    SOLUTION BY
•5 REM      E. HORMATS
•6 REM =====================================
•10 PRINT"MONTE CARLO METHOD"
•20 INPUT"[DOWN][DOWN][RIGHT][RIGHT]NUMBE
 R OF CYCLES WANTED";CY
•30 PRINT"[DOWN][DOWN][RIGHT][RIGHT]":P=0
•40 FOR I=1 TO CY
•50 X=RND(0): Y=RND(0)
•60 Z=X*X+Y*Y
•70 IF Z>1 THEN 90
•80 P=P+1
•90 PRINT"AT ";I"[UP]"
•100 NEXT I
•110 PI=4*P/CY
•120 PRINT"[DOWN][DOWN][RIGHT]VALUE OF [P
 I] AFTER"CY "TRIALS IS = ",PI
•130 PRINT"[DOWN][DOWN]ANOTHER TRY (Y/N)"
•140 GET A$: IF A$="Y" THEN 20
•150 IF A$="N" THEN END
•160 GOTO 140
```

Jim Speers applied a unique twist to the "circle in a square" approach. His program for the C-128 compares the time needed to paint one quarter of a square compared with the time needed to paint the inscribed quarter circle.

```
•1 REM =====================================
•2 REM    COMMODARES PROBLEM #49-4 :
•3 REM       EASY AS ...
•4 REM    SOLUTION BY
•5 REM       JIM SPEERS
•6 REM ===== FOR C-128 ONLY =============
•100 COLOR 0,1: COLOR 1,2: COLOR 4,12: GR
 APHIC 1,1: SCALE 1,1000,1000
•110 BOX 1,10,10,990,990: CIRCLE 1,10,990
 ,980,980,,90: TE=TI
•120 TB=TI: PAINT 1,20,980,1: TE=TI: TC=T
 E-TB
•130 TB=TI: PAINT 1,980,20,1: TE=TI: TT=T
 C+TE-TB: PY=4*TC/TT
•140 CHAR 1,4,2,"ANY KEY TO FIND THE VALU
 E OF PI",1
•150 GETKEY Z$
•160 COLOR 4,14: COLOR 5,14: COLOR 0,12
•170 GRAPHIC 0: PRINT"[CLEAR][DOWN][DOWN]
 [RVSON] APPROXIMATE VALUE OF PI IS [RVSO
 FF]": PRINT USING "[DOWN]#.[8"#"]";PY
•180 PRINT "[DOWN][RVSON] VARIATION FROM
 COMPUTER'S PI IS [RVSOFF]": PRINT USING
 "[DOWN]#.[8"#"]";ABS(PY-[PI])
```

Jim mentions that this approach comes pretty close to the calculus process of integration.

This short program from Jerry Nichols is based upon a formula for the cosine of a half angle when the full angle is known.

```
•1 REM =====================================
•2 REM    COMMODARES PROBLEM #49-4 :
•3 REM       EASY AS ...
•4 REM    SOLUTION BY
•5 REM       JERRY NICHOLS
•6 REM =====================================
•10 C=2[UPARROW].5/2: I=2[UPARROW].5*2: F
 OR X=1 TO 14: C=((1+C)/2)[UPARROW].5: I=
 I/C: PRINT I: NEXT X
```

In 14 iterations, the maximum internal accuracy of the computer is reached. Jerry sent a detailed analysis of this program. Send me a self-addressed, stamped envelope if you would like a copy.

Special credit goes to Craig Ewert, who sent a fairly lengthy program which includes its own math functions in order to calculate pi to greater accuracy than the computer can handle internally. He shows the value of pi as 3.141592653, which is accurate to eight decimal places. He did not state what the limitations of the program are. If you would like a listing of his program, send your request clearly stated to me with a self-addressed, stamped envelope.

Of course, there are always the "quick and dirty" solutions to problems like this. Shane Smart (Texarkana, AR) wasted very little computer time with this answer which gives full accuracy:

```
10 PRINT [PI] * 1
```

Jim Borden sent PI = 355/113 which he points out is accurate to within 2.667E−7 of the true value ("close enough for me," he said). This quotient is easy to remember if you write pairs of odd integers: 113355. Divide the last three digits by the first three. John Locke (Central Point, OR) sent this quick one:

```
10 PRINT 99023/31520
```

which gives pi as 3.14159264. Let's see. If we used that approximation for pi to calculate the volume of the earth, I wonder how far our answer would be from the actual value, assuming a perfect sphere. Looks like you have plenty to keep yourselves busy. See you next time. ☐

Congratulations to these readers not already mentioned this month:

Janet Bender (Vale, OR)
Kent Bowling
Mark Breault (Brandon, MAN)
Carlos Centeno
Tim Chapman (Cameron, IL)
Harlan Clussman
David Delong (Goshen, NY)
John Desclin (Brussels, Belgium)
Mark Dulski (Neenah, WI)
Kameron Hoffs (Phoenix, AZ)
Roy Kelly (Phillips, ME)
Keith Kushner
Paul Parker (Villa Rica, GA)
Donald Pellegrini
Chris Phillipi (Greenbelt, MD)
Raphael Richmond (Mentor, OH)
Mark Roschke (Chatsworth, CA)
Andrew Rosenthal (Flushing, NY)
C.J. Stewart (Phoenix, AZ)
John Velazquez (Chicago, IL)
James Whitman (Fredericton, NB)

PROGRAM LISTINGS

Attention new *Ahoy!* readers! You must read the following information very carefully prior to typing in programs listed in *Ahoy!* Certain Commodore characters, commands, and strings of characters and commands will appear in a special format. Follow the instructions and listings guide on this page.

On the following pages you'll find several programs that you can enter on your Commodore computer. But before doing so, read this entire page carefully.

To insure clear reproductions, *Ahoy!'s* program listings are generated on a daisy wheel printer, incapable of printing the commands and graphic characters used in Commodore programs. These are therefore represented by various codes enclosed in brackets [ ]. For example: the SHIFT CLR/HOME command is represented onscreen by a heart. The code we use in our listings is [CLEAR]. The chart below lists all such codes which you'll encounter in our listings, except for one other special case.

The other special case is the COMMODORE and SHIFT characters. On the front of most keys are two symbols. The symbol on the left is obtained by pressing that key while holding down the COMMODORE key; the symbol on the right, by pressing that key while holding down the SHIFT key. COMMODORE and SHIFT characters are represented in our listings by a lower-case "s" or "c" followed by the symbol of the key you must hit. COMMODORE J, for example, is represented by [c J], and SHIFT J by [s J].

Additionally, any character that occurs more than two times in a row will be displayed by a coded listing. For example, [3 "[LEFT]"] would be 3 CuRSoR left commands in a row, [5 "[s EP]"] would be 5 SHIFTed English Pounds, and so on. Multiple blank spaces will be noted in similar fashion: e.g., 22 spaces as [22 " "].

Sometimes you'll find a program line that's too long for the computer to accept (C-64 lines are a maximum of 80 characters, or 2 screen lines long; C-128 lines, a maximum of 160 characters, 2 or 4 screen lines in 40 or 80 columns respectively). To enter these lines, refer to the *BASIC Command Abbreviations Appendix* in your User Manual.

On the next page you'll find our *Bug Repellent* programs for the C-128 and C-64. The version for your machine will help you proofread programs after typing them. (Please note: the *Bug Repellent* line codes that follow each program line, in the whited-out area, should *not* be typed in. See instructions preceding each program.)

On the second page following you will find *Flankspeed*, our ML entry program, and instructions on its use.

Call *Ahoy!* at 212-239-6089 with any problems (if busy or no answer after three rings, call 212-239-0855).

WHEN YOU SEE	IT MEANS	YOU TYPE		YOU WILL SEE
[CLEAR]	Screen Clear	SHIFT	CLR/HOME	
[HOME]	Home		CLR/HOME	
[UP]	Cursor Up	SHIFT	↑ CRSR ↓	
[DOWN]	Cursor Down		↑ CRSR ↓	
[LEFT]	Cursor Left	SHIFT	← CRSR →	
[RIGHT]	Cursor Right		← CRSR →	
[SS]	Shifted Space	SHIFT	Space	
[INSERT]	Insert	SHIFT	INST/DEL	
[DEL]	Delete		INST/DEL	
[RVSON]	Reverse On	CNTRL	9	
[RVSOFF]	Reverse Off	CNTRL	0	
[UPARROW]	Up Arrow		↑	↑
[BACKARROW]	Back Arrow		←	←
[PI]	PI		π	π
[EP]	English Pound		£	£

WHEN YOU SEE	IT MEANS	YOU TYPE		YOU WILL SEE
[BLACK]	Black	CNTRL	1	
[WHITE]	White	CNTRL	2	
[RED]	Red	CNTRL	3	
[CYAN]	Cyan	CNTRL	4	
[PURPLE]	Purple	CNTRL	5	
[GREEN]	Green	CNTRL	6	
[BLUE]	Blue	CNTRL	7	
[YELLOW]	Yellow	CNTRL	8	
[F1]	Function 1		F1	
[F2]	Function 2	SHIFT	F1	
[F3]	Function 3		F3	
[F4]	Function 4	SHIFT	F3	
[F5]	Function 5		F5	
[F6]	Function 6	SHIFT	F5	
[F7]	Function 7		F7	
[F8]	Function 8	SHIFT	F7	

BUG REPELLENT FOR THE 64 & 128 By BUCK CHILDRESS

Please note: the *Bug Repellent* programs listed here are for *Ahoy!* programs published from the May 1987 issue onward! For older programs, use the older version.

Type in, save, and run *Bug Repellent*. You'll be asked if you want automatic saves to take place. If so, you're prompted for the device, DISK (D) or TAPE (T). You then pick a starting file number, 0 through 99. Next, you enter a name, up to 14 characters long. At this point, *Bug Repellent* verifies your entries and gives you a chance to change them if you want. If no changes are needed, *Bug Repellent* activates itself. (Pressing RETURN without answering the prompts defaults to disk drive and begins your files with "00BACKUP".)

Type NEW and begin entering an *Ahoy!* program. As you enter program lines and press RETURN, a *Bug Repellent* code appears at the top of your screen. If it doesn't match the code in the program listing, an error exists. Correct the line and the codes will match.

If used, automatic saves take place every 15 minutes. When the RETURN key is pressed on a program line, the screen changes color to let you know that a save will begin in about three seconds. You may cancel the save by pressing the RUN STOP key. The file number increments after each save. It resets to 00 if 99 is surpassed. After saving, or cancelling, the screen returns to its original color and the timer resets for 15 minutes.

When you've finished using *Bug Repellent*, deactivate it by typing SYS 49152 [RETURN] for the Commodore 64 or SYS 4864 [RETURN] for the Commodore 128.

C-64 BUG REPELLENT

```
·10 PRINTCHR$(147)"LOADING AND CHECKING THE DATA[3"."]":J
=49152
·20 FORB=0TO11:READA:IFA<0ORA>255THEN40
·30 POKEJ+B,A:X=X+A:NEXTB:READA:IFA=XTHEN50
·40 PRINT:PRINT"ERROR IN DATA LINE:"PEEK(64)*256+PEEK(63)
:END
·50 X=0:J=J+12:IFJ<49456THEN20
·60 POKE198,0:POKE49456,0:A$="Y":B$=A$:C$="D":D$="DISK":D
=8:PRINTCHR$(147)
·70 INPUT"DO YOU WANT AUTOMATIC SAVES (Y/N)";A$:PRINT:IFA
$="Y"THEN90
·80 PRINT"NO AUTOMATIC SAVES[3"."]":GOTO150
·90 POKE49456,1:INPUT"DISK OR TAPE (D/T)";C$:IFC$<>"D"THE
ND=1:D$="TAPE"
·100 POKE49457,D:D$=D$+" DRIVE":PRINT:INPUT"FILE NUMBER (
0-99)";N
·110 N$=RIGHT$(STR$(N),2):IFN<10THENN$=CHR$(48)+CHR$(N+48
)
·120 F$="BACKUP":PRINT:INPUT"FILENAME";F$:F$=N$+LEFT$(F$,
14):L=LEN(F$)
·130 POKE49458,L:FORJ=1TOL:POKE49458+J,ASC(MID$(F$,J,1)):
NEXTJ:PRINT
·140 PRINT"SAVING DEVICE ** "D$:PRINT"STARTING WITH ** "F
$
·150 PRINT:INPUT"IS THIS CORRECT (Y/N)";B$:IFB$<>"Y"THEN6
0
·160 POKE770,131:POKE771,164:SYS49152:END
·170 DATA169,79,32,210,255,162,38,160,192,204,3,3,1507
·180 DATA208,10,162,131,160,164,169,70,32,210,255,44,1615
·190 DATA169,78,32,210,255,142,2,3,140,3,3,76,1113
·200 DATA36,193,32,96,165,134,122,132,123,32,115,0,1180
·210 DATA170,240,243,162,255,134,58,144,3,76,150,164,1799
·220 DATA32,107,169,32,121,165,173,0,2,240,5,169,1215
·230 DATA79,141,2,3,76,162,164,169,0,133,2,133,1064
·240 DATA251,133,252,133,254,24,101,20,69,254,230,254,197
5
·250 DATA24,101,21,69,254,170,230,254,164,252,185,0,1724
·260 DATA2,133,253,201,34,208,6,165,2,73,255,133,1465
·270 DATA2,201,32,208,4,165,2,240,8,138,24,101,1125
·280 DATA253,69,254,170,44,198,254,230,252,164,253,208,23
49
·290 DATA213,138,41,240,74,74,74,74,24,105,129,141,1327
·300 DATA44,193,138,41,15,24,105,129,141,45,193,162,1230
·310 DATA0,189,43,193,240,12,157,0,4,173,134,2,1147
·320 DATA157,0,216,232,208,239,169,38,141,2,3,173,1578
·330 DATA48,193,240,23,165,161,201,212,176,4,165,160,1748
·340 DATA240,13,238,32,208,160,0,32,225,255,208,6,1617
·350 DATA32,33,193,76,38,192,232,208,242,200,208,239,1893
·360 DATA32,68,229,169,0,168,174,49,193,32,186,255,1555
·370 DATA173,50,193,162,51,160,193,32,189,255,169,43,1670
·380 DATA166,45,164,46,32,216,255,162,1,189,51,193,1520
·390 DATA168,200,152,201,58,144,2,169,48,157,51,193,1543
·400 DATA201,48,208,3,202,16,234,32,33,193,76,116,1362
·410 DATA164,206,32,208,169,0,170,168,76,219,255,160,1827
·420 DATA1,1,160,0,0,65,72,79,89,33,0,0,500
```

C-128 BUG REPELLENT

```
·10 PRINTCHR$(147)"LOADING AND CHECKING THE DATA[3"."]":J
=4864
·20 FORB=0TO11:READA:IFA<0ORA>255THEN40
·30 POKEJ+B,A:X=X+A:NEXTB:READA:IFA=XTHEN50
·40 PRINT:PRINT"ERROR IN DATA LINE:"PEEK(66)*256+PEEK(65)
:END
·50 X=0:J=J+12:IFJ<5213THEN20
·60 POKE208,0:POKE5213,0:A$="Y":B$=A$:C$="D":D$="DISK":D=
8:PRINTCHR$(147)
·70 INPUT"DO YOU WANT AUTOMATIC SAVES (Y/N)";A$:PRINT:IFA
$="Y"THEN90
·80 PRINT"NO AUTOMATIC SAVES[3"."]":GOTO150
·90 POKE5213,1:INPUT"DISK OR TAPE (D/T)";C$:IFC$<>"D"THEN
D=1:D$="TAPE"
·100 POKE5214,D:D$=D$+" DRIVE":PRINT:INPUT"FILE NUMBER (0
-99)";N
·110 N$=RIGHT$(STR$(N),2):IFN<10THENN$=CHR$(48)+CHR$(N+48
)
·120 F$="BACKUP":PRINT:INPUT"FILENAME";F$:F$=N$+LEFT$(F$,
14):L=LEN(F$)
·130 POKE5215,L:FORJ=1TOL:POKE5215+J,ASC(MID$(F$,J,1)):NE
XTJ:PRINT
·140 PRINT"SAVING DEVICE ** "D$:PRINT"STARTING WITH ** "F
$
·150 PRINT:INPUT"IS THIS CORRECT (Y/N)";B$:IFB$<>"Y"THEN6
0
·160 POKE770,198:POKE771,77:SYS4864:END
·170 DATA32,58,20,169,41,162,19,236,3,3,208,4,955
·180 DATA169,198,162,77,141,2,3,142,3,3,224,19,1143
·190 DATA208,7,32,125,255,79,78,0,96,32,125,255,1292
·200 DATA79,70,70,0,96,162,0,134,251,189,0,2,1053
·210 DATA240,19,201,48,144,9,201,58,176,5,133,251,1485
·220 DATA232,208,238,134,252,165,251,208,3,76,198,77,2042
·230 DATA169,0,166,235,164,236,133,253,133,254,142,47,193
2
·240 DATA20,140,48,20,24,101,22,69,254,230,254,24,1206
·250 DATA101,23,69,254,170,230,254,164,252,185,0,2,1704
·260 DATA133,251,201,34,208,6,165,253,73,255,133,253,1965
·270 DATA201,32,208,4,165,253,240,8,138,24,101,251,1625
·280 DATA69,254,170,44,198,254,230,252,164,251,208,213,23
07
·290 DATA138,41,240,74,74,74,74,24,105,65,141,88,1138
·300 DATA20,138,41,15,24,105,65,141,89,20,32,79,769
·310 DATA20,189,85,20,240,6,32,210,255,232,208,245,1742
·320 DATA174,47,20,172,48,20,24,32,240,255,173,93,1298
·330 DATA20,240,27,165,161,201,212,176,4,165,160,240,1771
·340 DATA17,32,65,20,238,32,208,238,1,214,32,225,1322
·350 DATA255,208,6,32,49,20,76,198,77,232,208,242,1603
·360 DATA200,208,239,32,66,193,173,95,20,162,96,160,1644
·370 DATA20,32,189,255,169,0,170,32,104,255,169,0,1395
·380 DATA174,94,20,168,32,186,255,169,45,174,16,18,1351
·390 DATA172,17,18,32,216,255,162,1,189,96,20,168,1346
·400 DATA200,152,201,58,144,2,169,48,157,96,20,201,1448
·410 DATA48,208,3,202,16,234,32,49,20,141,0,2,955
·420 DATA76,183,77,58,59,32,65,20,206,32,208,206,1222
·430 DATA1,214,169,0,170,168,76,219,255,32,79,20,1403
·440 DATA169,26,141,0,214,173,0,214,16,251,96,162,1462
·450 DATA0,142,0,255,96,19,18,32,32,32,32,146,804
·460 DATA0,1,0,0,65,72,79,89,33,0,0,0,339
```

FLANKSPEED FOR THE C-64 By GORDON F. WHEAT

Flankspeed will allow you to enter machine language *Ahoy!* programs without any mistakes. Once you have typed the program in, save it for future use. While entering an ML program with *Flankspeed* there is no need to enter spaces or hit the carriage return. This is all done automatically. If you make an error in a line a bell will ring and you will be asked to enter it again. To LOAD in a program Saved with *Flankspeed* use LOAD "name",1,1 for tape, or LOAD "name",8,1 for disk. The function keys may be used after the starting and ending addresses have been entered.

f1–SAVEs what you have entered so far.

f3–LOADs in a program worked on previously.

f5–To continue on a line you stopped on after LOADing in the previous saved work.

f7–Scans through the program to locate a particular line, or to find out where you stopped the last time you entered the program. It temporarily freezes the output as well.

```
·100 POKE53280,12:POKE53281,11                                   OP
·105 PRINT"[CLEAR][c 8][RVSON][15" "]FLANKSPEED[15" "]";         FP
·110 PRINT"[RVSON][5" "]MISTAKEPROOF ML ENTRY PROGRAM[6" "
 ]"                                                              JP
·115 PRINT"[RVSON][9" "]CREATED BY G. F. WHEAT[9" "]"            FA
·120 PRINT"[RVSON][3" "]COPR. 1987, ION INTERNATIONAL INC.
 [3" "]"                                                         AJ
·125 FORA=54272TO54296:POKEA,0:NEXT                              ND
·130 POKE54272,4:POKE54273,48:POKE54277,0:POKE54278,249:PO
 KE54296,15                                                      NP
·135 FORA=680TO699:READB:POKEA,B:NEXT                            FL
·140 DATA169,251,166,253,164,254,32,216,255,96                   FF
·145 DATA169,0,166,251,164,252,32,213,255,96                     EK
·150 B$="STARTING ADDRESS IN HEX":GOSUB430:AD=B:SR=B             KP
·155 GOSUB480:IFB=0THEN150                                       OE
·160 POKE251,T(4)+T(3)*16:POKE252,T(2)+T(1)*16                   AM
·165 B$="ENDING ADDRESS IN HEX":GOSUB430:EN=B                    PE
·170 GOSUB470:IFB=0THEN150                                       PG
·175 POKE254,T(2)+T(1)*16:B=T(4)+1+T(3)*16                       GM
·180 IFB>255THENB=B-255:POKE254,PEEK(254)+1                      HG
·185 POKE253,B:PRINT                                             EC
·190 REM  GET HEX LINE                                           ED
·195 GOSUB495:PRINT": [c P][LEFT]";:FORA=0TO8                    KD
·200 FORB=0TO1:GOTO250                                           IH
·205 NEXTB                                                       IJ
·210 A%(A)=T(1)+T(0)*16:IFAD+A-1=ENTHEN340                       FA
·215 PRINT" [c P][LEFT]";                                        EG
·220 NEXTA:T=AD-(INT(AD/256)*256):PRINT" "                       II
·225 FORA=0TO7:T=T+A%(A):IFT>255THENT=T-255                      GL
·230 NEXT                                                        GI
·235 IFA%(8)<>TTHENGOSUB375:GOTO195                              FL
·240 FORA=0TO7:POKEAD+A,A%(A):NEXT:AD=AD+8:GOTO195               IM
·245 REM    GET HEX INPUT                                        PA
·250 GETA$:IFA$=""THEN250                                        GA
·255 IFA$=CHR$(20)THEN305                                        GO
·260 IFA$=CHR$(133)THEN535                                       LM
·265 IFA$=CHR$(134)THEN560                                       IG
·270 IFA$=CHR$(135)THENPRINT" ":GOTO620                          HO
·275 IFA$=CHR$(136)THENPRINT" ":GOTO635                          HE
·280 IFA$>"@"ANDA$<"G"THENT(B)=ASC(A$)-55:GOTO295                MI
·285 IFA$>"/"ANDA$<":"THENT(B)=ASC(A$)-48:GOTO295                DJ
·290 GOSUB415:GOTO250                                            JA
·295 PRINTA$"[c P][LEFT]";                                       PK
·300 GOTO205                                                     FA
·305 IFA>0THEN320                                                BI
·310 A=-1:IFB=1THEN330                                           BB
·315 GOTO220                                                     FA
·320 IFB=0THENPRINTCHR$(20);CHR$(20);:A=A-1                      BF
·325 A=A-1                                                       FK
·330 PRINTCHR$(20);:GOTO220                                      PH
·335 REM LAST LINE                                               CP
·340 PRINT" ":T=AD-(INT(AD/256)*256)                             KH
·345 FORB=0TOA-1:T=T+A%(B):IFT>255THENT=T-255                    OD
·350 NEXT                                                        OB
·355 IFA%(A)<>TTHENGOSUB375:GOTO195                              LH
·360 FORB=0TOA-1:POKEAD+B,A%(B):NEXT                             BO
·365 PRINT:PRINT"YOU ARE FINISHED!":GOTO535                      MB
·370 REM BELL AND ERROR MESSAGES                                 LM
·375 PRINT:PRINT"LINE ENTERED INCORRECTLY":PRINT:GOTO415         JK
·380 PRINT:PRINT"INPUT A 4 DIGIT HEX VALUE!":GOTO415             PG
·385 PRINT:PRINT"ENDING IS LESS THAN STARTING!":B=0:GOTO41
 5                                                               IK
·390 PRINT:PRINT"ADDRESS NOT WITHIN SPECIFIED RANGE!":B=0:
 GOTO415                                                         HK
·395 PRINT:PRINT"NOT ZERO PAGE OR ROM!":B=0:GOTO415              DM
·400 PRINT"?ERROR IN SAVE":GOTO415                               JK
·405 PRINT"?ERROR IN LOAD":GOTO415                               IO
·410 PRINT:PRINT:PRINT"END OF ML AREA":PRINT                     JO
·415 POKE54276,17:POKE54276,16:RETURN                            BF
·420 OPEN15,8,15:INPUT#15,A,A$:CLOSE15:PRINTA$:RETURN            DH
·425 REM GET FOUR DIGIT HEX                                      IM
·430 PRINT:PRINTB$;:INPUTT$                                      OL
·435 IFLEN(T$)<>4THENGOSUB380:GOTO430                            JD
·440 FORA=1TO4:A$=MID$(T$,A,1):GOSUB450:IFT(A)=16THENGOSUB
 380:GOTO430                                                     AK
·445 NEXT:B=(T(1)*4096)+(T(2)*256)+(T(3)*16)+T(4):RETURN         KB
·450 IFA$>"@"ANDA$<"G"THENT(A)=ASC(A$)-55:RETURN                 GM
·455 IFA$>"/"ANDA$<":"THENT(A)=ASC(A$)-48:RETURN                 NJ
·460 T(A)=16:RETURN                                              IC
·465 REM ADDRESS CHECK                                           OL
·470 IFAD>ENTHEN385                                              HO
·475 IFB<SRORB>ENTHEN390                                         LE
·480 IFB<256OR(B>40960ANDB<49152)ORB>53247THEN395                OB
·485 RETURN                                                      HE
·490 REM ADDRESS TO HEX                                          PM
·495 AC=AD:A=4096:GOSUB520                                       AP
·500 A=256:GOSUB520                                              NF
·505 A=16:GOSUB520                                               LG
·510 A=1:GOSUB520                                                HE
·515 RETURN                                                      JD
·520 T=INT(AC/A):IFT>9THENA$=CHR$(T+55):GOTO530                  OC
·525 A$=CHR$(T+48)                                               JI
·530 PRINTA$;:AC=AC-A*T:RETURN                                   AA
·535 A$="**SAVE**":GOSUB585                                      IC
·540 OPEN1,T,1,A$:SYS680:CLOSE1                                  AB
·545 IFST=0THENEND                                               FB
·550 GOSUB400:IFT=8THENGOSUB420                                  PM
·555 GOTO535                                                     FI
·560 A$="**LOAD**":GOSUB585                                      PE
·565 OPEN1,T,0,A$:SYS690:CLOSE1                                  PO
·570 IFST=64THEN195                                              OI
·575 GOSUB405:IFT=8THENGOSUB420                                  CO
·580 GOTO560                                                     GN
·585 PRINT" ":PRINTTAB(14)A$                                     KA
·590 PRINT:A$="":INPUT"FILENAME";A$                              IO
·595 IFA$=""THEN590                                              HK
·600 PRINT:PRINT"TAPE OR DISK?":PRINT                            HL
·605 GETB$:T=1:IFB$="D"THENT=8:A$="@0:"+A$:RETURN                NP
·610 IFB$<>"T"THEN605                                            KO
·615 RETURN                                                      PH
·620 B$="CONTINUE FROM ADDRESS":GOSUB430:AD=B                    DD
·625 GOSUB475:IFB=0THEN620                                       NK
·630 PRINT:GOTO195                                               HN
·635 B$="BEGIN SCAN AT ADDRESS":GOSUB430:AD=B                    FK
·640 GOSUB475:IFB=0THEN635                                       LN
·645 PRINT:GOTO670                                               HI
·650 FORB=0TO7:AC=PEEK(AD+B):GOSUB505:IFAD+B=ENTHENAD=SR:G
 OSUB410:GOTO195                                                 LM
·655 PRINT" ";:NEXTB                                             LE
·660 PRINT:AD=AD+8                                               CD
·665 GETB$:IFB$=CHR$(136)THEN195                                 JD
·670 GOSUB495:PRINT": ";:GOTO650                                 KE
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 67 and 68 explain these codes and provide other essential information on entering ***Ahoy!*** programs. Refer to these pages **before** entering any programs!

RASHGAR
FROM PAGE 16

```
•10 COLOR0,1:COLOR4,1:COLOR1,1:GRAPHIC1,1
 :GRAPHIC.:BANK15:WIDTH1:PRINTCHR$(19)CHR
 $(19)CHR$(147)CHR$(11)CHR$(142):GOSUB20:
 GOTO80                                   MH
•20 PRINTTAB(8)"[WHITE]AHOY! [CYAN]MAGAZI
 NE PRESENTS[3"."][3"[DOWN]"]"            MG
•30 PRINTTAB(9)"[CYAN][c R][s C][s I][s U
 ][s C][s I][s U][s C][c W][c S] [c A][s
 U][s C][s I][s U][s C][s I][c R][s C][s
 I]"                                      OA
•40 PRINTTAB(9)"[c 7][c Q][s C][c W][c Q]
 [s C][c W][s J][s C][s I][c Q][s C][c W]
 [s B][c A][c S][c Q][s C][c W][c Q][s C]
 [c W]"                                   JE
•50 PRINTTAB(9)"[BLUE][c E] [c E][c E] [c
  E][c Q][s C][s K][c X] [c Z][s J][s C][
 s K][c E] [c E][c E] [c E]":PRINTTAB(9)"
 [GREEN]MASTER OF DIMENSION X[BLUE]"      KO
•60 PRINTTAB(13)"[DOWN]FOR THE C-128":PRI
 NTTAB(14)"[DOWN](C)1988 CMB"             OC
•70 PRINTTAB(7)"[3"[DOWN]"]BY CLEVELAND M
 . BLAKEMORE":PRINTTAB(9)"[DOWN]JOYSTICK
 IN PORT TWO!":RETURN                     JF
•80 FORJ=4864TO5313:READA$:POKEJ,DEC(A$):
 NEXT:FORA=3072TO3140:READA$:POKEA,DEC(A$
 ):NEXT:POKE253,3                         MO
•90 DIMB(8),S$(8),G$(8),C(4):FORJ=1TO8:B(
 J)=2[UPARROW](J-1):NEXT:C(.)=14:C(1)=4:C
 (2)=15:C(3)=7:C(4)=12:GOSUB380:GOSUB360  KD
•100 U=1:T=2:G=15:C=.:K=360:S=1:H=3:F=4:I
 =8:P=8184:D=45:X=159:Y=100:DX=12:DY=40:M
 X=511:MY=255:MR=3:Q=8:L=1:B=76:PUDEF"0":
 F$="[5"#"]"                              EM
•110 J=5:GOSUB420:SC=.                    MA
•120 DEFFNC(X)=((J>.ANDA=.)AND(CANDF))OR(
 CANDU)                                   EC
•130 DEFFNR(X)=RND(U)*(F+U)+F:DEFFNK(X)=R
 ND(U)*K                                  JM
•140 DEFFNX(X)=(RSPPOS(H,.)-DX)ANDMX:DEFF
 NY(X)=(RSPPOS(H,U)-DY)ANDMY              CC
•150 GRAPHICU,U:GOSUB400:COLORU,G:CHARU,.
 ,.," 1":CHARU,DY-T,.," 3"                PE
•160 FORJ=UTOI:MOVSPRJ,.#.:NEXT           DD
•170 POKEP,B:POKEP+U,B+F:POKEP+T,B+F+U:MO
 VSPRU,170,135:MOVSPRT,170,135:SPRITEU,.,
 F+T,U,.,.,.,.:SPRITET,.,I+H,.,.,.,.,.:MOVSPR
 H,170,70:SPRITEH,.,G+U,.,.,.,.,.         PD
•180 FORJ=FTOI:SPRITEJ,.,T,.,.,.,.,.:MOVSPR
 J,MY+Y,MY:POKE(P-U)+J,56:A=(J-T)*55      LA
•190 MOVSPRJ,A#U:NEXT:POKE53248+21,255:SY
 S3072                                    BP
•200 FORJ=.TOG:C=BUMP(U):NEXT             PH
•210 REM MAIN LOOP                        LA
•220 DO:J=JOY(T):A=JANDG:IFATHENMOVSPRH,I
 ;(A-U)*D:ELSEIFJTHENSPRITET,U,T:DRAWU,X,
 YTOFNX(.),FNY(.):SPRITET,U,G:DRAW.,X,YTO
 FNX(.),FNY(.):SOUNDU,P,G,T,MX,MX,H       BM
•230 MOVSPRFNR(.),FNK(.)#S:C=BUMP(U):LOOP
 UNTILFNC(.)                              PF
•240 IF(CANDU)=.THEN310:ELSEA=U:POKEP,B:F
 ORJ=.TOI:GRAPHICH,.:A=A+U+(A=16)*G:SPRIT
 EU,.:GRAPHICU,.:SPRITEU,U,A:MOVSPRT,168+
 K,132+K:NEXT:SYS65418                    ON
•250 FORJ=.TOT:SPRITEJ+U,.,F+T:POKEP+J,82
 :NEXT:FORJ=.TOD:A=RND(U)*H+U:MOVSPRA,170
 ,135:MOVSPRA,RND(U)*K#I:SOUNDU,RND(U)*P,
 I,U,,,H:SPRITEA,U:NEXT                   AO
•260 SPRITEU,.:SPRITET,.:SPRITEH,.:MR=MR-
 U:A$=STR$(MR):CHARU,DY-T,.,A$:IFMRTHEN16
 0                                        LN
•270 FORJ=UTOI:MOVSPRJ,.#.:MOVSPRJ,400,.:
 SPRITEJ,.,T,.,U,U,.:SPRSAVG$(J),J:POKE(P
 -U)+J,55+J:NEXT:IFSC>HITHENHI=SC         OK
•280 FORJ=UTOF:SPRITEJ,U:SOUNDU,J*1000,G,
 T,J*MX,J*900,T,MX*T:FORA=400TO60+J*24STE
 P-F:MOVSPRJ,A,Y:NEXT:NEXT                BN
•290 FORJ=5TOI:SPRITEJ,U:SOUNDU,J*1000,G,
 T,J*MX,J*900,T,MX*T:FORA=400TO34+J*24STE
 P-F:MOVSPRJ,A,Y+D:NEXT:NEXT:J=1:GOSUB430
 :COLOR1,4                                GK
•300 FORJ=UTOI:SPRITEJ,.:SPRSAVS$(J),J:NE
 XT:GRAPHIC.:GOTO100                      MK
•310 SPRITEH,U,T:POKEP+T,82:FORJ=HTOI:IFC
 =(B(J)ORF)THENBEGIN:SPRITEJ,.:MOVSPRJ,MY
 +X,MY:SPRITEJ,U:SC=SC+Y*L:A$=STR$(SC)+"
 ":CHARU,G+T,.,A$                         AF
•320 Q=Q-U:IFQ=.THENQ=I:L=L+U:A$=STR$(L):
 CHARU,.,.,.,A$:SOUNDH,P*T,G,T,MX,P,T:IF(L=
 F)ORL=ITHENS=S+U                         KK
•330 BEND                                 GE
•340 NEXT:POKEP+T,81:SPRITEH,U,G+U:GOTO20
 0                                        BL
•350 REM CREATE AGGRESSOR SHAPES          FP
•360 FORJ=1TO8:BOX0,0,0,24,21,0,1:BOX1,9,
 9,14,14,(9-J)*11:BOX1,7,7,16,16,J*11:BOX
 1,5,5,18,18,(9-J)*11:SSHAPES$(J),1,1,24,
 21:SPRSAVS$(J),J:NEXT:RETURN             FK
•370 REM CREATE GAME OVER CHARACTERS      LK
•380 GRAPHIC2,.,0:FORJ=1TO8:BOX0,0,0,24,2
 1,0,1:A$=MID$("GAMEOVER",J,1):CHAR.,1,1,
 A$:SSHAPEG$(J),1,1,24,21:SPRSAVG$(J),J:N
 EXT:GRAPHIC.:RETURN                      FA
•390 REM SHIELD ENERGIZE                  AH
•400 C=.:FORA=20TO180STEP35:COLOR1,C(C):C
 =C+U:SOUND2,A*60,450,1,A*5,A*20,.:FORJ=.
 TO10STEP2:CIRCLE1,158,95,A+J:NEXTJ,A:RET
 URN                                      KM
•410 REM FLASH PROMPT AND WAIT FOR FIRE   NN
```

```
•420 CHAR.,.,22:PRINT"[c 7]HIGH SCORE [WH
ITE]";:PRINTUSINGF$;HI;:PRINTSPC(5)"[BLU
E]CURRENT SCORE [WHITE]";:PRINTUSINGF$;S
C                                        LF
•430 IFJOY(T)>127THEN430                 CH
•440 A=U:DO:CHARU,10,20,"DEPRESS FIRE BUT
TON!":A=A+U+(A=16)*16:COLORJ,A:LOOPUNTIL
JOY(2)>127:RETURN                        FE
•450 REM RASHGAR SPRITE DATA             DJ
•460 DATA 7F,FF,FE,FF,FF,FF,E3,FF,E3,E1,F
F,C3,E0,FF,83,E0,7F,3,F0,3E,7,F8,7F,F,FF
,FF,FF,7F,EB,FF,3F,EB,FE,1F,EB           EK
•470 DATA FC,1F,FF,FC,1F,80,7C,1F,,3C,1E,
,1C,F,,38,7,80,70,3,FF,E0,1,FF,C0,,FF,80
,,7F,FF,FE,FF,FF,FF                      CB
•480 DATA E3,FF,E3,E1,FF,C3,E0,FF,83,E0,7
F,3,F0,3E,7,F8,7F,F,FF,FF,FF,7F,EB,FF,3F
,EB,FE,1F,EB,FC,1F,FF,FC,1F,FF           BG
•490 DATA FC,1F,80,7C,1E,,1C,F,80,78,7,FF
,F0,3,FF,E0,1,FF,C0,,FF,80,,7F,FF,FE,FF,
FF,FF,E3,FF,E3,E1,FF,C3                  HP
•500 DATA E0,FF,83,E0,7F,3,F0,3E,7,F8,7F,
F,FF,FF,FF,7F,EB,FF,3F,EB,FE,1F,EB,FC,1F
,FF,FC,1F,FF,FC,1F,E1,FC,1F,0            MO
•510 DATA 3C,F,E1,F8,7,FF,F0,3,FF,E0,1,FF
,C0,,FF,80,,7F,FF,FE,FF,FF,FF,E3,FF,E3,E
1,FF,C3,E0,FF,83,E0,7F,3                 KF
•520 DATA F0,3E,7,F8,7F,F,FF,FF,FF,7F,EB,
FF,3F,EB,FE,1F,EB,FC,1F,FF,FC,1F,FF,FC,1
F,FF,FC,1F,80,7C,F,FF,F8,7,FF            DP
•530 DATA F0,3,FF,E0,1,FF,C0,,FF,80,,,,,,
,,,,,,,,,,,6,,60,6,,60,,,0               HL
•540 DATA ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,0                                     AM
•550 DATA ,,,,,,,,,,,,,1F,FF,F8,10,18,8,
10,18,8,10,,8,10,,8,10,,8,1F,,F8         GF
•560 DATA 1F,,F8,10,,8,10,,8,10,,8,10,18,
8,10,18,8,1F,FF,F8,,,,,,,,,,,,,80        HG
•570 DATA 08,,50,21,24,,A8,48,14,8,90,8A,
29,24,1,2A,40,12,94,A1,9,69,40,84,DE,88,
13,AB,,3,55,6A,AE,AB,80                  ME
•580 DATA 03,56,40,4,A9,22,49,7C,90,2,86,
40,29,21,24,2,4,90,84,20,40,1,4,22,20,24
,10,,,0                                  NB
•590 REM INTERRUPT DATA                  KK
•600 DATA 78,A9,15,8D,14,3,A9,C,8D,15,3,A
9,4,85,FB,A9,7,85,FC,58,60,C6,FB,D0,29,A
9,4,85,FB,A5,FC,18,69,38,A2,5,9D,FA,1F,C
A,D0,FA,C6,FC,10,4,A9,7,85               PJ
•610 DATA FC,C6,FD,10,4,A9,3,85,FD,A9,4C,
18,65,FD,8D,F8,1F,4C,65,FA               IH
```

PHOBIA

FROM PAGE 20

```
•10 POKE56,62:CLR                        GB
•20 PRINTCHR$(147):V=53248:POKEV+32,11:PO
KEV+33,.:POKEV+21,.                      PF
•30 FORX=.TO37STEP3:POKE1024+X,64+32:POKE
1024+X+1,128+32:POKE1024+X+2,192+32      FO
•40 POKE1984+X,192+32:POKE1984+X+1,128+32
:POKE1984+X+2,64+32                      HH
•50 NEXT                                 LE
•60 FORX=.TO22STEP3                      PB
•70 POKE1024+X*40,192+32:POKE1024+(X+1)*4
0,128+32:POKE1024+(X+2)*40,64+32         CC
•80 POKE1063+X*40,64+32:POKE1063+(X+1)*40
,128+32:POKE1063+(X+2)*40,192+32         IJ
•90 NEXT:POKE2023,128+32                 CC
•100 POKEV+17,PEEK(V+17)OR64:POKEV+34,1:P
OKEV+35,15:POKEV+36,11                   JI
•110 GOSUB1400:FORX=.TO319:READA:POKE248*
64+X,A:NEXT                              DH
•120 FORX=.TO128:POKE253*64+X,.:NEXT:POKE
253*64+30,168:POKE253*64+33,84           JA
•130 FORX=.TO2:POKE254*64+X,255:NEXT:FORX
=.TO63:POKE255*64+X,PEEK(248*64+X):NEXT  ML
•140 A=PEEK(V+34):B=PEEK(V+35):C=PEEK(V+3
6)                                       FF
•150 POKEV+34,C:POKEV+35,A:POKEV+36,B    HA
•160 IFPEEK(56320)AND16THENFORX=.TO30:NEX
T:GOTO140                                MM
•170 GOSUB1500                           PN
•180 REM INITIALIZATION                  GK
•190 DIMV,X,J,JP,VM,SB,P,W,E,G,N,T,VC,F,A
,B,C,Q,R,L,U,D,H,Z,SC                    OH
•200 DIMC(1),P$(1),C$(25),A$(25),F$(1),P(
1),B(8)                                  NI
•210 JP=56320:VM=2040:S=70:A=.:B=1:C=1:R=
119:L=123:M=127:U=4:D=2:SB=248:H=8       JD
•220 G=16:P=147:W=147:E=255:N=126:T=125:Y
=.5:C(.)=55378:C(1)=55578:LV=.:VC=V+30   BA
•230 P(.)=472:P(1)=484:SID=54272:F$(1)="[
5"[DOWN]"]"                              JF
•240 P$(.)="[WHITE][sEP][c 5][RVSON][4" "
][DOWN][5"[LEFT]"]":PC$(.)="[RVSON][sEP]
[4"[c T]"][DOWN][5"[LEFT]"][5"[c T]"]"   AL
•250 P$(1)="[RVSON][c 5][4" "][WHITE][RVS
OFF][c *][DOWN][5"[LEFT]"]":PC$(1)="[RVS
ON][4"[c T]"][c *][DOWN][5"[LEFT]"][5"[c
 T]"]"                                   FJ
•260 B$="":FORX=.TO28:B$=B$+CHR$(32):NEXT
:FORX=.TO8:B(X)=2[UPARROW]X:NEXT         IJ
•270 L$="YOUR LIFE FORCE IS WEAK[3"."]YOU
 LACK SUFFICIENT WILL TO SURVIVE[3"."]"  JO
•280 W$="CONGRATULATIONS, YOU HAVE GRADUA
TED FROM THE INSTITUTE OF REALITY[3"."]" MK
•290 FORX=.TO12:READC$(X*2),A$(X*2),C$(X*
2+1),A$(X*2+1):NEXT                      EO
•300 POKEV,S:POKEV+1,W:POKEVM,248:POKEV+3
9,0                                      HA
•310 POKEVM+1,253:POKEV+2,255:POKEV+3,148
:POKEV+40,13:POKEV+23,12:POKEV+29,2      EE
•320 POKEV+4,15:POKEV+16,4:POKEV+5,130:PO
KEV+6,.:POKEV+7,130:POKEV+41,.           NK
```

```
·330 POKEV+8,107:POKEV+10,221:POKEV+9,166
 :POKEV+11,166                               JM
·340 POKEVM+4,254:POKEVM+5,254:POKEV+43,.
 :POKEV+44,.:POKEV+27,48                     OG
·350 POKEV+42,.:POKEVM+2,251:POKEVM+3,252
 :POKEV+21,13OR(((2[UPARROW]LV)AND482)>.)
 *-2                                         DP
·360 A=9:FORX=.TO1:IFP(X)ANDB(LV)THENGOSU
 B660:POKEV+21,PEEK(V+21)ORB(4+X)            CN
·370 A=24:PRINT"[HOME]"TAB(87)F$(X)B$:POK
 EC(X),7:NEXT                                PA
·380 Z=INT(RND(1)*13):PRINT"[HOME][c 7][D
 OWN][DOWN]";TAB(20-LEN(C$(Z*2))/2)C$(Z*2
 )                                           LD
·390 PRINT"[4"[DOWN]"]";TAB(20-LEN(C$(Z*2
 +1))/2)C$(Z*2+1)                            LE
·400 POKE1858,49+LV:F=-1:A=-(LV>4)+-(LV>7
 ):GOSUB700                                  HA
·410 POKESID+6,233:POKESID+13,234:POKESID
 +20,234:FORX=.TO10:J=PEEK(VC):NEXT          PD
·420 REM MAIN LOOP                           HH
·430 J=PEEK(JP):ON-(S>E)GOTO520:POKEV,S:O
 N-(PEEK(V+D)=E)GOSUB740                     AL
·440 IFJ<>MTHENS=S+(J=R)*-U+(J=L)*U:C=C+Y
 :POKEVM,SB+C:IFC=DORC=.THENY=-Y             NM
·450 IF(JANDG)=.ANDK=.THENK=-D:Q=.:P=P+K:
 GOSUB760                                    JN
·460 IFKTHENP=P+K:Q=Q+B:POKEV+B,P:IFQ=HTH
 ENK=-K                                      OJ
·470 IFJ=NORJ=TTHENF=N-J:POKEC(F),U+B:POK
 EC(B-F),D:GOSUB720                          GP
·480 K=K*-(P<>W):POKEV+D,(PEEK(V+D)-U)AND
 E:POKEV+U+D,(PEEK(V+U+D)+B+A)ANDE           GD
·490 X=PEEK(VC):ON(XANDB)+BGOTO430:IFX<>H
 +BTHENF=-B:GOTO600                          HK
·500 IFS<>ETHENS=PEEK(V+U+D)+G+B:X=PEEK(V
 C):GOTO430                                  KH
·510 F=-B:GOTO600                            DH
·520 IFF<.THENS=E-U:GOTO430                  NG
·530 IFA$(Z*2+F)<>""THEN600                  DI
·540 POKEV+1,W:GOSUB700:GOSUB850:FORX=15T
 O100:POKEV+U,X:FORA=.TO30:NEXTA,X           HP
·550 POKEV+16,PEEK(V+16)OR1:FORX=.TO100ST
 EP2:FORA=.TO30:NEXT:POKEV,X:POKEVM,SB+C     CA
·560 C=C+Y:IFC=DORC=.THENY=-Y                LP
·570 NEXT:LV=LV+1:POKEV+21,.:S=70:K=.:P=W
 :GOSUB680:SC=SC+Z*50+LV*250:A$=STR$(SC)     PI
·580 POKE214,19:PRINT:PRINTTAB(4)"[RVSON]
 [c 5]";:GOSUB970:PRINT:ON-(LV<9)GOTO300:
 GOSUB910                                    FI
·590 S$="":S$=S$+B$+B$+W$+B$:PRINT"[c 6]"
 :GOTO630                                    GE
·600 POKEVM,E:GOSUB780:FORX=.TO63:POKEV+3
 9,XAND15:POKEE*64+X,.:NEXT                  FF
·610 IFF=-1THENS$="":S$=S$+B$+B$+L$+B$:F=
 .:PRINT"[c 6]":GOTO630                      NK
·620 S$="":S$=S$+B$+B$+A$(Z*2+F)+"!"+B$      GC
·630 GOSUB700:FORX=1TOLEN(S$):GOSUB720       EI
·640 FORA=.TO15:NEXT:PRINT"[HOME]"TAB(87)
 F$(F)MID$(S$,X,29):NEXT:X=FRE(.):RUN        DB
·650 REM CREATE PITS                         AO
·660 POKE214,13:PRINT:PRINTTAB(A)P$(X);:P
 OKE646,RND(1)*5+2:PRINTPC$(X):RETURN        IG
·670 REM ERASE PITS                          PJ
·680 POKE214,13:PRINT:FORX=1TO30:PRINTTAB
 (X)"[RVSON][WHITE] [DOWN][LEFT][c 8] [DO
 WN][LEFT] [UP][UP]";:NEXT:PRINT:RETURN      AM
·690 REM CLEAR SID CHIP                      KG
·700 FORX=.TO23:POKESID+X,.:NEXT:POKESID+
 24,15:RETURN                                HN
·710 REM CLICK                               LB
·720 POKESID+1,20:POKESID+4,17:POKESID+4,
 16:RETURN                                   GA
·730 REM LASER FIRE                          BG
·740 POKESID+8,10:POKESID+11,129:POKESID+
 11,128:RETURN                               BO
·750 REM JUMP                                LJ
·760 POKESID+15,5:POKESID+18,33:POKESID+1
 8,32:RETURN                                 BL
·770 REM VAPORIZED                           DG
·780 POKESID+14,67:POKESID+15,5              CH
·790 POKESID+4,21:POKESID+18,17              EO
·800 FORA=10TO50                             PG
·810 FORC=.TOESTEPA*5:POKESID+1,C:POKESID
 +15,E-A*5:POKEV+39,CAND15:NEXT:NEXT         OD
·830 POKESID+4,20:POKESID+18,16:RETURN       AF
·840 REM DOOR SLIDING FORWARD                JK
·850 POKESID+5,13*16+15:POKESID+6,15*16+1
 5                                           ID
·860 POKESID+15,3:POKESID+14,220             EP
·870 POKESID+1,23:POKESID,10:POKESID+4,21    ON
·880 FORX=.TO2000:NEXT:FORX=15TO1STEP-.05
 :POKESID+24,X:NEXT                          AP
·890 POKESID+4,.:FORX=1TO500:NEXT:POKESID
 +24,15:RETURN                               NN
·900 REM FANFARE                             BC
·910 A$="251331421502421502":GOSUB700:POK
 ESID+6,96                                   AJ
·920 FORX=1TO18STEP3                         EK
·930 POKESID+1,VAL(MID$(A$,X,2)):POKESID+
 4,17                                        PE
·940 FORA=1TO90*VAL(MID$(A$,X+2,1)):NEXT     PJ
·950 POKESID+4,16:NEXT:RETURN                EJ
·960 REM RIGHT JUSTIFIED SCORE               AE
·970 PRINTRIGHT$("[5"0"]"+(RIGHT$(A$,LEN(
 A$)-1)),5);:X=FRE(.):RETURN                 EP
·980 REM SPRITE DATA                         GK
·990 DATA0,112,,,232,,,252                   LG
·1000 DATA0,,176,,1,216,,7                   FN
·1010 DATA224,,14,240,,29,248,128            HI
·1020 DATA59,255,192,103,239,192,112,32      CP
·1030 DATA0,103,224,,7,224,,7                GC
·1040 DATA176,,15,188,,15,62,.               FG
·1050 DATA30,15,,60,7,128,120,3              MG
·1060 DATA216,224,1,240,240,,224,.           KP
·1070 DATA0,112,,,232,,,252                  AL
```

```
·1080 DATA0,,176,,1,216,,7                    OG
·1090 DATA224,,14,240,,29,208,.               PN
·1100 DATA59,208,,59,208,,62,24               BP
·1110 DATA0,13,248,,3,240,,3                  IG
·1120 DATA176,,3,176,,3,176,.                 IC
·1130 DATA3,176,,3,176,,7,112                 DH
·1140 DATA0,7,56,,3,156,,.                    AK
·1150 DATA0,56,,,116,,,126                    CP
·1160 DATA0,,88,,,236,,1                      LE
·1170 DATA240,,3,252,,5,222,48                MM
·1180 DATA013,239,240,29,243,224,120,.        JI
·1190 DATA0,115,240,,2,248,,3                 JE
·1200 DATA60,,3,222,,7,143,.                  FG
·1210 DATA15,7,128,30,3,192,60,1              KN
·1220 DATA236,112,,248,120,,48,.              AM
·1230 DATA0,,,,31,128,,55                     PJ
·1240 DATA128,,119,128,,247,128,1             IK
·1250 DATA247,128,3,247,128,3,247,128         OK
·1260 DATA3,247,128,3,247,128,3,247           EF
·1270 DATA128,3,247,128,3,247,128,3           DB
·1280 DATA247,128,3,247,128,1,247,128         OB
·1290 DATA0,247,128,,119,128,,55              AH
·1300 DATA128,,31,128,,,,.                    MC
·1310 DATA0,,,1,248,,1,236                    NH
·1320 DATA0,1,238,,1,239,,1                   NI
·1330 DATA239,128,1,239,192,1,239,192         EA
·1340 DATA1,239,192,1,239,192,1,239           AL
·1350 DATA192,1,239,192,1,239,192,1           JN
·1360 DATA239,192,1,239,192,1,239,128         CK
·1370 DATA1,239,,1,238,,1,236                 CG
·1380 DATA0,1,248,,,,,.                       ED
·1390 REM TITLE                               HI
·1400 PRINT"[HOME][DOWN][DOWN]"TAB(9)"[BL
UE]AHOY! MAGAZINE PRESENTS[DOWN]"             GB
·1410 PRINTTAB(6)"[WHITE][RVSON][4":"][RV
SOFF] [RVSON]:[RVSOFF]  [RVSON]:[RVSOFF]
 [RVSON][4":"][RVSOFF] [RVSON][4":"][RVS
OFF] [RVSON][3":"][RVSOFF] [RVSON][4":"]
"                                             NC
·1420 PRINTTAB(6)"[RVSON]:[RVSOFF]  [RVSO
N]:[RVSOFF] [RVSON]:[RVSOFF]  [RVSON]:[R
VSOFF] [RVSON]:[RVSOFF]  [RVSON]:[RVSOFF
] [RVSON]:[RVSOFF]  [RVSON]:[RVSOFF]  [R
VSON]:[RVSOFF]  [RVSON]:[RVSOFF]  [RVSON
]:"                                           OJ
·1430 PRINTTAB(6)"[RVSON][4":"][RVSOFF] [
RVSON][4":"][RVSOFF] [RVSON]:[RVSOFF]  [
RVSON]:[RVSOFF] [RVSON][4":"][RVSOFF]  [
RVSON]:[RVSOFF]  [RVSON][4":"]"               OP
·1440 PRINTTAB(6)"[RVSON]:[RVSOFF][4" "][
RVSON]:[RVSOFF]  [RVSON]:[RVSOFF] [RVSON
]:[RVSOFF]  [RVSON]:[RVSOFF] [RVSON]:[RV
SOFF]  [RVSON]:[RVSOFF]  [RVSON]:[RVSOFF
]  [RVSON]:[RVSOFF]  [RVSON]:"                OC
·1450 PRINTTAB(6)"[RVSON]:[RVSOFF][4" "][
RVSON]:[RVSOFF]  [RVSON]:[RVSOFF] [RVSON
][4":"][RVSOFF] [RVSON][4":"][RVSOFF] [R
VSON][3":"][RVSOFF] [RVSON]:[RVSOFF]  [R
VSON]:"                                       OI
·1460 PRINTTAB(11)"[DOWN][DOWN][YELLOW]BY
 CLEVE BLAKEMORE":PRINTTAB(16)"[DOWN][DO
WN](C) 1988"                                  KC
·1470 PRINTTAB(10)"[DOWN][DOWN]JOYSTICK I
N PORT TWO"                                   EL
·1480 PRINTTAB(5)"[WHITE][3"[DOWN]"]PRESS
 FIRE BUTTON TO START GAME":RETURN            NG
·1490 REM GAME SCREEN                         DB
·1500 PRINTCHR$(147);:POKEV+17,PEEK(V+17)
AND191                                        GD
·1510 FORX=.TO9:FORA=.TO39:PRINT"[RVSON][
c 8] ";:NEXT:NEXT                             PI
·1520 PRINT"[RVSOFF][c 4][c *][RVSON]";:F
ORX=.TO37:PRINT" ";:NEXT:PRINT"[RVSOFF][
sEP]";                                        PM
·1530 FORX=.TO2:PRINT"[RIGHT]";:FORA=.TO3
7:PRINT"[RVSON][c 5] ";:NEXT:PRINT"[RIGH
T]";:NEXT                                     FM
·1540 PRINT"[WHITE][RVSON][sEP]";:FORX=.T
037:PRINT" ";:NEXT:PRINT"[c *]";              HC
·1550 FORX=.TO8:FORA=.TO39:PRINT"[RVSON][
c 8] ";:NEXT:NEXT                             GG
·1560 FORX=.TO39:POKE1984+X,160:POKE56256
+X,15:NEXT                                    CK
·1570 FORX=217TO242:POKEX,PEEK(X)OR128:NE
XT                                            PH
·1580 PRINT"[HOME][DOWN][RVSOFF]";:FORX=.
TO1:PRINTTAB(5)"[c 4][c *][RVSON][31" "]
[RVSOFF][sEP]"                                DF
·1590 PRINTTAB(5)" [c G][29" "][c M] "        EB
·1600 PRINTTAB(5)"[WHITE][RVSON][sEP][31"
 "][c *][DOWN][DOWN]":NEXT                    LH
·1610 POKE214,17:PRINT:PRINTTAB(13)"[c 4]
[c *][RVSON][13" "][RVSOFF][sEP]"             EK
·1620 FORX=.TO2:PRINTTAB(13)" [RVSON][c 5
][13" "][RVSOFF] ":NEXT                       DC
·1630 PRINTTAB(13)"[WHITE][RVSON][sEP][13
" "][c *]":PRINTTAB(15)"[3"[UP]"][c 5][R
VSON]P H O B I A"                             LE
·1640 POKE214,17:PRINT:PRINTTAB(4)"[RVSON
][c 8]SCORE":GOSUB1690                        DI
·1650 POKE214,17:PRINT:PRINTTAB(32);:A$="
[c 7]1":GOSUB1680                             KG
·1660 PRINT"[HOME]";:FORX=.TO1:A$="[YELLO
W][s Q]":GOSUB1680:NEXT                       LM
·1670 RETURN                                  BK
·1680 PRINT"[DOWN][RIGHT][c 4][c *][RVSON
] [RVSOFF][sEP][DOWN][3"[LEFT]"][c M]"A$
"[c 4][c G][DOWN][3"[LEFT]"][RVSON][WHIT
E][sEP] [c *][DOWN]":RETURN                   MK
·1690 PRINTTAB(2)"[c 4][c *][RVSON][7" "]
[RVSOFF][sEP]":PRINTTAB(2)" [RVSON][c 5]
 [5"0"] [RVSOFF] "                            LF
·1700 PRINTTAB(2)"[RVSON][WHITE][sEP][7"
"][c *]":RETURN                               MB
·1710 REM DOOR DATA                           JF
·1720 DATAWATER,YOU ARE NOT A WATER BREAT
```

```
HING CREATURE,AIR,""                        NN
·1730 DATAEARTH,,FIRE,EARTH BURIES BUT FI
RE INCINERATES                              OD
·1740 DATAWOLVES,"DRAGONS ARE FANTASY-BUT
 WOLVES ARE REAL",DRAGONS,""                EG
·1750 DATASLOW DEATH,"YOU CAN SURVIVE TOR
TURE,BUT NEVER DEATH",SLOW TORTURE,""       OE
·1760 DATAREAL,,UNREAL,YOU CANNOT EXIST O
UTSIDE REALITY                              MC
·1770 DATADESPAIR,,PANIC,PANIC IS LOSS OF
 CONSCIOUS CONTROL AND CAUSES DEATH         KE
·1780 DATALIFE & DEATH,,ETERNAL CONFINEME
NT,LIFE IS BETTER THAN IMPRISONMENT         DJ
·1790 DATASAFE CHAOS,CHAOS IS OUTSIDE OF
REALITY,DANGEROUS ORDER,""                  DF
·1800 DATADEMONS,,TIGERS,"DEMONS ARE NOT
REAL,TIGERS ARE"                            LG
·1810 DATANIGHTMARES,"",RATTLESNAKES,VENO
M KILLS BUT NIGHTMARES ARE HARMLESS         AL
·1820 DATADEATH SENTENCE,"",EXTERMINATION
,"WHERE THERE'S LIFE,THERE'S HOPE"          BL
·1830 DATASEARING HEAT,,ABSOLUTE ZERO COL
D,NOTHING CAN SURVIVE ABSOLUTE ZERO         DJ
·1840 DATAACCELERATION,TO BE ACCELERATED
TOO QUICKLY IS FATAL,DEACCELERATION,""      PH
```

MOB BLOB
FROM PAGE 41

Starting address in hex: C000
Ending address in hex: C202
SYS to start: 49152

***Flankspeed* required for entry! See page 69.**

```
C000: 78 A9 1F 8D 14 03 A9 C0 50
C008: 8D 15 03 A9 F8 8D 12 D0 C0
C010: AD 11 D0 29 7F 8D 11 D0 B7
C018: A9 81 8D 1A D0 58 60 AD 22
C020: 19 D0 8D 19 D0 29 01 D0 7C
C028: 07 AD 0D DC 58 4C 31 EA 87
C030: AD 00 DC 29 0F 8D A7 02 2A
C038: A9 0F 38 ED A7 02 A8 B9 23
C040: DA C1 F0 0F 8D A8 02 A8 BD
C048: B9 BE C1 2D EF C1 F0 03 55
C050: 20 64 C0 20 C0 C0 20 14 6B
C058: C1 A9 F8 8D 12 D0 4C BC 36
C060: FE 4C 31 EA AD A8 02 0A 2A
C068: AA BD C7 C1 30 19 F0 2B BF
C070: AD 00 D0 18 6D D9 C1 8D 9D
C078: 00 D0 90 1F A9 01 4D 10 01
C080: D0 8D 10 D0 4C 9B C0 AD 16
C088: 00 D0 38 ED D9 C1 8D 00 A8
C090: D0 B0 08 A9 01 4D 10 D0 F2
C098: 8D 10 D0 AD A8 02 0A AA 14
C0A0: E8 BD C7 C1 30 0F F0 17 18
C0A8: AD 01 D0 18 6D D9 C1 8D D6
C0B0: 01 D0 4C BF C0 AD 01 D0 CE
C0B8: 38 ED D9 C1 8D 01 D0 60 3A
C0C0: AD 10 D0 29 01 D0 17 AD 0F
C0C8: 00 D0 38 CD EA C1 B0 06 03
C0D0: AD EA C1 8D 00 D0 AD EC 24
C0D8: C1 D0 17 4C E3 C0 AD EC 0E
C0E0: C1 F0 0F AD 00 D0 38 CD 27
C0E8: EB C1 90 06 AD EB C1 8D 16
C0F0: 00 D0 AD 01 D0 38 CD ED 35
C0F8: C1 B0 09 AD ED C1 8D 01 60
C100: D0 4C 13 C1 AD 01 D0 38 A9
C108: CD EE C1 90 06 AD EE C1 7B
C110: 8D 01 D0 60 A2 07 AD F0 18
C118: C1 3D BF C1 F0 41 8A 0A 5F
C120: A8 BD F1 C1 18 6D 00 D0 90
C128: 99 00 D0 90 0C BD BF C1 6E
C130: 0D 10 D0 8D 10 D0 4C 44 1D
C138: C1 A9 FF 5D BF C1 2D 10 BF
C140: D0 8D 10 D0 C8 BD F9 C1 C1
C148: 18 6D 01 D0 99 00 D0 A9 B3
C150: 01 2D 10 D0 F0 09 BD BF D6
C158: C1 0D 10 D0 8D 10 D0 CA 41
C160: D0 B4 60 A9 2C 20 FF AE EA
C168: 20 EB B7 A5 14 8D EB C1 21
C170: A5 15 8D EC C1 20 9B B7 DA
C178: 8E EA C1 60 20 9B B7 8E 16
C180: ED C1 20 9B B7 8E EE C1 E2
C188: 60 20 9B B7 8E F0 C1 60 FD
C190: 20 9B B7 8E EF C1 60 20 C4
C198: 9B B7 8E A7 02 20 9B B7 97
C1A0: 8A AE A7 02 9D F1 C1 60 35
C1A8: 20 9B B7 8E A7 02 20 9B 10
C1B0: B7 8A AE A7 02 9D F9 C1 A4
C1B8: 60 20 9B B7 8E D9 C1 01 B7
C1C0: 02 04 08 10 20 40 80 00 BF
C1C8: 00 00 FF 01 FF 01 00 01 CB
C1D0: 01 00 01 FF 01 FF 00 FF D3
C1D8: FF 01 00 01 05 00 07 08 EE
C1E0: 06 00 03 02 04 00 00 00 EF
C1E8: 00 00 18 F8 00 32 D2 FF FE
C1F0: FE 00 18 30 48 00 18 30 C8
C1F8: 48 00 00 00 00 15 15 15 80
C200: 15 00 FF 15
```

SNAP SNAKE
FROM PAGE 43

```
·10 GOTO30                                  PL
·20 POKEI+D,C:POKEG,F+D:POKEG,F:RETURN      AI
·30 POKE53281,.:POKE53280,.:POKE53282,8:P
OKE53283,11:PRINT"[CLEAR]"CHR$(8)CHR$(14
2)                                          BE
·40 GOSUB890:FORJ=50176TO50602:READL:POKE
```

```
 J,L:NEXT                                   AB
•50 FORJ=.TO63:READL:POKE51712+J,L:NEXT    LN
•60 DIM M(8),H(8),L(200),E(47)             BK
•70 DIMJ,L,C,V,D,P,T,F,I,G,K,L,U,B,S,EG,S
 C,LV,SS,OF                                 MN
•80 FORX=1TO8:READM(X):NEXT:FORX=1TO8:REA
 DH(X):NEXT                                 EP
•90 FORX=.TO15:READJ:POKE53248+X,J:NEXT    MM
•100 FORX=.TO7:POKE50168+X,40:C=-(C=.):PO
 KE53248+39+X,6+3*C:NEXT                    DP
•110 FORX=.TO47:READE(X):NEXT:SF$="[HOME]
 ":FORX=.TO23:SF$=SF$+"[DOWN]":NEXT         OC
•120 SYS50176                               BF
•130 GOSUB610:WAIT56320,16,16:GOSUB410      JA
•140 J=.:C=.:T=.:P=.:I=54272:G=I+4:V=5324
 8:D=1:F=128:K=200:E=49151:B=32:S=81        IM
•150 LV=.:SC=.:OF=6144:SS=3                 JJ
•160 POKEV+27,255:POKEV+23,.:POKEV+29,.:P
 OKEV+28,.:POKEV+16,.                       AJ
•170 DEFFNT(X)=J>.ANDM(J)ANDC<KANDT>EANDP
 EEK(T)<=B                                  CB
•180 DEFFNC(X)=J=.ANDP=B                    LG
•190 GOSUB790:EG=LV*4+D:POKEV+21,102+(LVA
 ND1)*51:POKEI+6,233:POKEI+20,233           CH
•200 POKEV+34,(PEEK(V+34)+1)AND15:POKE501
 37,177+LV:POKE681,9-LV                     LG
•210 FORX=.TO10:J=PEEK(V+31):NEXT           BG
•220 C=.:L=49611:POKEL,H(D):L(.)=L          NL
•230 REM MAIN LOOP                          OE
•240 J=USR(.):T=L+M(J)                      DP
•250 IFFNT(.)THENPOKEL,S:L=T:C=C+D:L(C)=L
 :P=PEEK(T):POKEL,H(J):GOSUB20              AL
•260 IFUSR(D)=.ANDCTHENPOKEL(C),B:C=C-D:L
 =L(C):POKEL,H(D):GOSUB20                   FM
•270 J=PEEK(V+B-D):ON-(FNC(.))GOTO240       JJ
•280 IFJTHEN370                             CL
•290 IFP=B-2THENEG=EG-D:SC=SC+50            CL
•300 IFP=B-4THENSC=SC+100*(LV+D)            DJ
•310 IFP=B-DTHENSC=SC+INT(RND(D)*5)*200+2
 00                                         GB
•320 P=B:POKEL+OF,15:POKEI+15,P:POKEI+18,
 F+D:POKEI+18,F                             ID
•330 A$=STR$(SC):PRINTSF$"[RVSON][GREEN][
 RIGHT][RIGHT]"RIGHT$("[5"0"]"+(RIGHT$(A$
 ,LEN(A$)-D)),5);                           DG
•340 ON-(EG>.)GOTO240:IFLV<8THENLV=LV+D     ND
•350 IFC>.THENPOKEL(C),B:C=C-D:L=L(C):POK
 EL,H(D):GOSUB20:GOTO350                    DB
•360 GOTO190                                JH
•370 FORJ=CTO.STEP-D:POKEL(J),B:FORX=.TO2
 :POKEV+35,X:NEXT:NEXT                      CG
•380 POKE50151-SS,160:POKEV+35,11:SS=SS-D
 :C=.:ON-(SS>.)GOTO210                      MI
•390 FORX=.TO2000:NEXT:POKEV+21,.:GOTO130   PC
•400 REM GAME SCREEN                        AL
•410 PRINT"[c 8][CLEAR][DOWN][RIGHT][c 8]
 [6"[s V]"][RIGHT][s V][3"[RIGHT]"][3"[s
 V]"][RIGHT][3"[s V]"][RIGHT][7"[s V]"][3
 "[RIGHT]"][s V][s V][RIGHT][4"[s V]"][RI
 GHT][s V]"                                 LK
•420 PRINT"[RIGHT][s V][6"[RIGHT]"][s V][
 5"[RIGHT]"][3"[s V]"][RIGHT][s V][RIGHT]
 [s V][RIGHT][s V][RIGHT][s V][RIGHT][s V
 ][3"[RIGHT]"][s V][RIGHT][s V][RIGHT][3"
 [s V]"][RIGHT][s V]"                       MH
•430 PRINT"[RIGHT][s V][RIGHT][4"[s V]"][
 RIGHT][s V][3"[RIGHT]"][s V][5"[RIGHT]"]
 [s V]"SPC(13)"[s V][5"[RIGHT]"][s V]"      KG
•440 PRINT"[RIGHT][3"[s V]"][RIGHT][4"[s
 V]"][3"[RIGHT]"][6"[s V]"][RIGHT][8"[s V
 ]"][3"[RIGHT]"][s V][s V][RIGHT][6"[s V]
 "][DOWN][DOWN]"                            OD
•450 PRINT"[RIGHT][8"[s V]"][3"[RIGHT]"][
 c 6][7"[s V]"][RIGHT][7"[s V]"][3"[RIGHT
 ]"][c 8][6"[s V]"][RIGHT][s V][s V]"       BC
•460 PRINT"[RIGHT][s V][6"[RIGHT]"][s V][
 3"[RIGHT]"][c 6][s V][s V][BLUE][5"#"][R
 IGHT][5"#"][c 6][s V][s V][3"[RIGHT]"][c
  8][s V][6"[RIGHT]"][s V][s V]"            PI
•470 PRINT"[RIGHT][s V][RIGHT][4"[s V]"][
 RIGHT][s V][3"[RIGHT]"][c 6][s V][s V][B
 LUE][4"#"][sEP][RIGHT][c *][4"#"][c 6][s
  V][s V][3"[RIGHT]"][c 8][s V][RIGHT][6"
 [s V]"]"                                   OL
•480 PRINT"[3"[RIGHT]"][s V][4"[RIGHT]"][
 s V][3"[RIGHT]"][c 6][s V][s V][BLUE][3"
 #"][sEP][3"[RIGHT]"][c *][3"#"][c 6][s V
 ][s V][3"[RIGHT]"][c 8][s V][RIGHT][s V]
 [4"[RIGHT]"][s V][s V]"                    KN
•490 PRINT"[RIGHT][s V][RIGHT][s V][RIGHT
 ][4"[s V]"]"SPC(21)"[s V][RIGHT][s V][RI
 GHT][3"[s V]"][RIGHT][s V]"                GN
•500 PRINT"[RIGHT][s V][RIGHT][3"[s V]"][
 RIGHT][s V][4"[RIGHT]"][c 6][s V][s V][B
 LUE][3"#"][RVSON][c *][RVSOFF][3"[RIGHT]
 "][RVSON][sEP][RVSOFF][3"#"][c 6][s V][s
  V][3"[RIGHT]"][c 8][s V][5"[RIGHT]"][s
 V][RIGHT][s V]"                            KB
•510 PRINT"[RIGHT][s V][RIGHT][3"[s V]"][
 RIGHT][s V][s V][3"[RIGHT]"][c 6][s V][s
  V][BLUE][4"#"][RVSON][c *][RVSOFF][RIGH
 T][RVSON][sEP][RVSOFF][4"#"][c 6][s V][s
  V][3"[RIGHT]"][c 8][7"[s V]"][RIGHT][s
 V]"                                        BL
•520 PRINT"[RIGHT][s V][RIGHT][s V]"SPC(8
 )"[c 6][s V][s V][BLUE][5"#"][RIGHT][5"#
 "][c 6][s V][s V]"SPC(11)"[c 8][s V]"      EE
•530 PRINT"[RIGHT][s V][s V][RIGHT][5"[s
 V]"][3"[RIGHT]"][c 6][7"[s V]"][RIGHT][7
 "[s V]"][3"[RIGHT]"][c 8][9"[s V]"][DOWN
 ][DOWN]"                                   EG
•540 PRINT"[RIGHT][s V][RIGHT][s V][s V][
 RIGHT][s V][RIGHT][s V][3"[RIGHT]"][s V]
 [RIGHT][s V][RIGHT][s V][s V][RIGHT][3"[
 s V]"][RIGHT][s V][s V][RIGHT][s V][3"[R
 IGHT]"][s V][s V][RIGHT][s V][RIGHT][s V
```

```
][RIGHT][s V][s V]"                         AH
•550 PRINT"[RIGHT][s V][RIGHT][s V][s V][
RIGHT][s V][RIGHT][s V][3"[RIGHT]"][s V]
[s V][RIGHT][s V][RIGHT][s V][s V][4"[RI
GHT]"][s V][s V][RIGHT][s V][3"[RIGHT]"]
[4"[s V]"][RIGHT][RIGHT][s V][RIGHT][s V
]"                                          AF
•560 PRINT"[RIGHT][RIGHT][5"[s V]"][RIGHT
][s V][3"[RIGHT]"][s V][5"[RIGHT]"][4"[s
 V]"][RIGHT][s V][s V][RIGHT][s V][6"[RI
GHT]"][4"[s V]"][RIGHT][s V]"               AB
•570 PRINT"[RIGHT][s V][6"[RIGHT]"][s V][
3"[RIGHT]"][s V][s V][RIGHT][s V][s V][R
IGHT][s V][4"[RIGHT]"][s V][RIGHT][RIGHT
][s V][3"[RIGHT]"][s V][s V][6"[RIGHT]"]
[s V]"                                      GC
•580 PRINT"[RIGHT][8"[s V]"][3"[RIGHT]"][
s V][s V][RIGHT][s V][RIGHT][s V][RIGHT]
[4"[s V]"][RIGHT][3"[s V]"][3"[RIGHT]"][
s V][RIGHT][7"[s V]"][DOWN]"                HN
•590 PRINT"[GREEN][RVSON] [5"0"]  [3"[RIG
HT]"] SNAP SNAKE (1)[3"[RIGHT]"]LIVES [3
"[s W]"] ";                                 HK
•600 PRINT"[UP]":PRINT"[RVSON][INSERT] ";
:PRINT"[HOME]";:RETURN                      EO
•610 REM TITLE SCREEN                       DE
•620 L$(1)="[BLUE][RIGHT][RVSON][3" "][DO
WN][3"[LEFT]"] [DOWN][LEFT][RVSOFF][c Y]
[c Y][RVSON] [RVSOFF][DOWN][3"[LEFT]"][3
"[c Y]"][3"[UP]"]"                          FE
•630 L$(2)="[PURPLE][RIGHT][RVSON] [c *]
[DOWN][3"[LEFT]"] [RVSOFF][c *][RVSON] [
DOWN][3"[LEFT]"] [RIGHT] [RIGHT][DOWN][4
"[LEFT]"][RVSOFF][c Y] [c Y][3"[UP]"]"      ON
•640 L$(3)="[BLUE][RIGHT][RVSON][3" "][DO
WN][3"[LEFT]"] [RIGHT] [DOWN][3"[LEFT]"]
 [RVSOFF][c Y][RVSON] [RVSOFF][DOWN][3"[
LEFT]"][c Y] [c Y][3"[UP]"]"                AN
•650 L$(4)="[PURPLE][RIGHT][RVSON] [RIGHT
] [DOWN][3"[LEFT]"] [RIGHT] [DOWN][3"[LE
FT]"] [RVSOFF][c Y][c J][DOWN][3"[LEFT]"
][c Y] [c Y][3"[UP]"]"                      IE
•660 L$(5)="[BLUE][RIGHT][RVSON][3" "][DO
WN][3"[LEFT]"] [DOWN][LEFT] [RVSOFF][c Y
][RVSON][DOWN][LEFT][LEFT][RVSOFF][3"[c
Y]"][3"[UP]"]"                              GL
•665 L$(6)="[c 5][RIGHT][RVSON]  [c *][DO
WN][3"[LEFT]"] [RIGHT] [DOWN][3"[LEFT]"]
 [RVSOFF][c Y][c Y][DOWN][3"[LEFT]"][c Y
][RIGHT][RIGHT][3"[UP]"]"                   KE
•670 PRINT"[CLEAR][DOWN]"TAB(7);:FORX=1TO
3:PRINTL$(X);:NEXT:PRINTL$(6)"[3"[DOWN]"
]"                                          OG
•680 PRINTTAB(10);:FORX=1TO5:PRINTL$(X);:
NEXT:PRINT"[4"[DOWN]"]"                     ID
•690 PRINTTAB(9)"[BLUE][s U][19"[s C]"][s
 I]"                                        MF
•700 PRINTTAB(9)"[s B][WHITE][UPARROW] [B
LUE]= [CYAN]10 POINTS"SPC(6)"[BLUE][s B]
"                                           PE
•710 PRINTTAB(9)"[s B][GREEN][EP] [BLUE]=
 [CYAN]50 POINTS*LEVEL[BLUE][s B]"          EB
•720 PRINTTAB(9)"[s B][PURPLE][BACKARROW]
 [BLUE]= [CYAN]MYSTERY?"SPC(7)"[BLUE][s
B]"                                         JD
•730 PRINTTAB(9)"[BLUE][s J][19"[s C]"][s
 K]"                                        NC
•740 IFSC>HITHENHI=SC                       OG
•750 IFHITHENPRINTTAB(11)"[YELLOW]HIGH SC
ORE :[WHITE]"HI                             ND
•760 PRINTTAB(8)"[WHITE][DOWN][DOWN]BY CL
EVELAND M. BLAKEMORE":PRINTTAB(14)"[DOWN
]FOR THE C-64"                              HK
•770 PRINTTAB(5)"[DOWN][BLUE]PRESS FIRE B
UTTON TO BEGIN GAME":RETURN                 BI
•780 REM DISTRIBUTE RANDOM EGGS             FI
•790 FORJ=.TOLV*4                           PM
•800 X=RND(D)*48:C=PEEK(E(X)):IFC=BORC=28
THENPOKEE(X),30:POKEE(X)+OF,1:GOTO820       FD
•810 GOTO800                                EL
•820 NEXT                                   LJ
•830 FORJ=.TO3                              NP
•840 X=RND(D)*48:C=PEEK(E(X)):IFC=BORC=28
THENPOKEE(X),28:POKEE(X)+OF,5:GOTO860       BI
•850 GOTO840                                HP
•860 NEXT:X=RND(D)*48:IFPEEK(E(X))=BTHENP
OKEE(X),31:POKEE(X)+OF,7                    HI
•870 RETURN                                 PH
•880 REM COPY CHAR SET                      DK
•890 POKE56333,127:POKE1,51                 KB
•900 POKE781,9:POKE782,1:POKE90,.:POKE91,
216:POKE88,.:POKE89,248:SYS41964            GG
•910 POKE1,55:POKE56333,129                 GF
•920 READJ:IFJ=-1THEN940                    EA
•930 FORL=.TO7:READC:POKE61440+J*8+L,C:NE
XT:GOTO920                                  KL
•940 POKE56578,PEEK(56578)OR3:POKE56576,P
EEK(56576)AND252                            HM
•950 POKE53272,PEEK(53272)AND15:POKE648,1
92:PRINTCHR$(147)                           KN
•960 POKE53272,(PEEK(53272)AND240)OR12:PO
KE53270,PEEK(53270)OR16:RETURN              GF
•970 REM CHARACTER DATA                     BF
•980 DATA028,60,189,90,60,255,60,126,153    DA
•990 DATA030,,24,60,126,126,126,126,60      OP
•1000 DATA031,60,66,189,165,165,189,66,60   PL
•1010 DATA035,219,54,205,54,205,54,219,10
8                                           EE
•1020 DATA065,40,170,170,170,235,170,170,
40                                          HD
•1030 DATA081,40,170,170,170,170,170,170,
40                                          FM
•1040 DATA083,40,170,170,235,170,170,170,
40                                          AI
•1050 DATA086,214,85,181,85,91,85,109,85    AP
•1060 DATA088,40,170,186,170,170,186,170,
```

```
40                                        AP
•1070 DATA090,40,170,174,170,170,174,170,
40                                        FE
•1080 DATA129,129,129,153,129,153,153,153
,255                                      DK
•1090 DATA139,24,17,3,7,3,17,24,255       PL
•1100 DATA142,25,9,1,33,49,57,57,255      JP
•1110 DATA147,129,129,159,129,249,129,129
,255                                      HM
•1120 DATA215,195,221,253,131,127,115,187
,195,-1                                   JF
•1130 REM INTERRUPT DATA                  MI
•1140 DATA120,169,32,141,20,3,169,196,141
,21,3,169,187,141,17,3,169                GD
•1150 DATA196,141,18,3,173,169,2,141,170,
2,32,229,196,88,96,206,170                MH
•1160 DATA2,240,3,76,181,196,173,169,2,14
1,170,2,169,7,141,167,2                   HE
•1170 DATA172,167,2,185,144,197,240,6,141
,168,2,32,68,196,76,173,196               CA
•1180 DATA173,168,2,10,170,189,126,197,8,
152,10,170,40,48,26,240,45                OM
•1190 DATA185,152,197,24,125,,208,157,,20
8,144,33,185,118,197,77,16                FK
•1200 DATA208,141,16,208,76,130,196,189,,
208,56,249,152,197,157,,208               EG
•1210 DATA176,9,185,118,197,77,16,208,141
,16,208,173,168,2,10,170,232              OI
•1220 DATA189,126,197,8,152,10,170,232,40
,48,15,240,23,185,152,197,24              LG
•1230 DATA125,,208,157,,208,76,172,196,18
9,,208,56,249,152,197,157                 LL
•1240 DATA,208,96,206,167,2,48,3,76,51,19
6,32,12,197,76,49,234                     NI
•1250 DATA32,170,177,152,41,1,170,208,22,
173,,220,41,15,133,253,169                GK
•1260 DATA15,56,229,253,168,185,160,197,1
68,169,,108,5,,173,,220                   CF
•1270 DATA41,16,168,169,,108,5,,162,24,16
9,,157,,212,202,16                        EO
•1280 DATA250,169,15,141,24,212,169,,141,
176,2,169,240,141,13,212,169              MO
•1290 DATA17,141,11,212,169,25,141,9,212,
141,10,212,96,169,68,133,250              OI
•1300 DATA169,197,133,251,206,175,2,208,3
3,169,20,141,175,2,172,176,2              KJ
•1310 DATA56,192,48,144,5,160,,140,176,2,
177,250,141,8,212,200,177                 HB
•1320 DATA250,141,7,212,200,140,176,2,173
,175,2,24,105,100,141,9,212               JC
•1330 DATA96,4,48,6,71,4,48,6,71,4,180,4,
48,6,167,6,167                            PC
•1340 DATA6,71,4,48,6,71,4,180,6,71,4,180
,4,251,4,251,4                            KM
•1350 DATA48,4,180,4,251,4,251,4,48,6,71,
4,180,4,48,4,48                           CL
•1360 DATA1,2,4,8,16,32,64,128,,,,255,1,2
55,1,,1                                   EA
•1370 DATA1,,1,255,1,255,,255,255,1,5,1,5
,3,7,3,7                                  GI
•1380 DATA1,1,1,1,1,1,1,1,,1,5,,7,8,6,,3,
2,4                                       BF
•1390 REM RAT DATA                        PA
•1400 DATA,,,,,,,.                        FA
•1410 DATA,,,,,,,.                        IK
•1420 DATA,,,,,,,.                        HE
•1430 DATA,,,112,124,,153,255             EH
•1440 DATA,14,254,192,21,125,80,52        II
•1450 DATA186,88,52,90,88,59,255,184      AF
•1460 DATA61,187,120,60,146,120,24,124    CB
•1470 DATA48,28,56,112,42,16,168,.        OD
•1480 REM SNAKE MOVEMENT DATA             HD
•1490 DATA -40,,1,,40,,-1,.               EK
•1500 REM SNAKE'S HEAD DIRECTIONS         KC
•1510 DATA 65,,88,,83,,90,.               HP
•1520 REM INITIAL SPRITE POSITIONS        EA
•1530 DATA 97,,97,,241,,241,,,83,,83,,171
,,171                                     FJ
•1540 REM EGG LOCATION DATA               PO
•1550 DATA49274,49279,49316,49249,49330,4
9207,49253                                NJ
•1560 DATA49255,49257,49263,49344,49224,4
9265,49596                                KP
•1570 DATA49714,49640,49638,49716,49755,4
9588,49667                                OK
•1580 DATA49550,49629,49914,49917,49994,4
9953,49885                                OC
•1590 DATA49887,49890,49931,50011,50016,4
9965,49926                                JD
•1600 DATA49928,50009,50048,50050,49931,5
0011,50016                                AH
•1610 DATA50055,49949,49904,49947,49908,5
0063                                      ND
```

RIGHT ON TRACK
FROM PAGE 32

DIRECTORY SCANNER

```
•1 REM==================================== NM
•2 REM                                     JD
•3 REM       DIRECTORY  SCANNER            FH
•4 REM                                     JF
•5 REM       RUPERT REPORT #54             CF
•6 REM                                     JH
•7 REM==== C-64/128 ==== 1541/1571 ====== MH
•10 REM                                    JL
•20 REM  READ AND DISPLAY DIRECTORY        DO
•30 REM  USING DOS DIRECT ACCESS COMMANDS  KC
•40 REM                                    LJ
•100 GOSUB 1000       :REM INITIALIZE      DD
```

```
·110 OPEN 15,DVC,15  :REM COMMAND CHANNEL LB
·120 GOSUB 7000     :REM CHECK DISK STATUS ME
·130 :REM   OPEN DATA CHANNEL 2; SET UP A
  DISK BUFFER FOR IT                      JB
·140 OPEN 1,DVC,2,"#"                     GL
·150 GOSUB 7000     :REM CHECK DISK STATUS AK
·160 :REM -- READ SECTOR INTO CHANNEL 2'S
  BUFFER IN DISK DRIVE :                  AA
·170 PRINT# 15,RD$;2;DVC;TK;SC            IL
·180 GOSUB 7000     :REM CHECK DISK STATUS BA
·190 GOSUB 2000      :REM READ DISK BUFFER
 INTO COMPUTER                            FA
·200 :REM -- INTERPRET DATA BYTES --      HC
·210 FOR FIL=0 TO 7                       AC
·220 GOSUB 3000 :REM DISPLAY DIR ENTRIES  NP
·230 REM >>> GOSUB 5000 :REM SELECT UNDEL
 ETION                                    KH
·240 NEXT FIL                             EG
·250 PRINT L$                             AE
·260 REM >>>> IF UDFLG THEN GOTO 6000:REM
 UNDELETE & END                           MN
·270 GOSUB 4000 :REM CONTINUE OR END?     HL
·280 GOTO 170   :REM CONTINUE             HG
·290 REM -------------------------------- LA
·1000 REM ========  INITIALIZE  ========= GG
·1010 PRINT CHR$(147); "==DIRECTORY SCANN
 ER==": PRINT                             PD
·1020 FALSE=0: TRUE=NOT FALSE: Z$=CHR$(0) MK
·1030 NUL$=Z$: SP$="  ": L$="--"          JH
·1040 FOR N=1 TO 4: NUL$=NUL$+NUL$: SP$=S
 P$+SP$: L$=L$+L$: NEXT                   CL
·1050 SP$="[UP]  " + SP$ + "  [UP]"       MD
·1060 FOR N=0 TO 4: READ TYP$(N): NEXT N  EE
·1070 DATA DEL, SEQ, PRG, USR, REL        FI
·1080 RD$="U1": WR$="U2"                  KK
·1090 TK=18: SC=1 :REM 1ST DIR TRK/SECTOR HA
·1100 DVC=8                               BL
·1110 RETURN                              OI
·1120 REM ------------------------------- AF
·2000 REM ========  READ SECTOR ========= FC
·2010 B$=""                               DN
·2020 PRINT "READING TRACK, SECTOR :"; TK
 ; SC                                     LN
·2030 PRINT "[33"="]"                     KO
·2040 PRINT "FILE NAME[7"."]TYPE:  TK SC
 LEN"                                     KA
·2050 PRINT "[33"="]"                     AN
·2060 PRINT# 15,"B-P";2;0   :REM POINT TO
 BYTE 0 OF CHNL 2 BUFFER                  CA
·2065 :REM GET NEXT DIR TRACK & SECTOR    HC
·2070 GET# 1,TK$: IF TK$="" THEN TK$=Z$   MA
·2080 GET# 1,SC$: IF SC$="" THEN SC$=Z$   AJ
·2090 FOR N=2 TO 255                      CM
·2100 GET# 1,D$                           LJ
·2110 IF D$="" THEN D$=Z$                 HP
·2120 B$=B$+D$   :REM READ BUFFER INTO B$ IB
·2130 NEXT N                              CK
·2140 RETURN                              PC
```

```
·2150 REM ------------------------------- HD
·3000 REM =======  INTERPRET DATA  ====== EN
·3010 PTR=FIL*32+1                        DH
·3020 FF=ASC(MID$(B$,PTR,1))    :REM FILE
 STATUS                                   JJ
·3030 MD=FF AND 192     :REM CLOSURE MODE MA
·3040 TYP=FF AND 7      :REM FILE TYPE    GD
·3050 :REM -- FIRST TK & SECTOR OF FILE   BJ
·3060 FTK=ASC(MID$(B$,PTR+1,1))           EF
·3070 FSC=ASC(MID$(B$,PTR+2,1))           LB
·3080 NM$=MID$(B$,PTR+3,16) :REM FILENAME ML
·3090 NULFLG=FALSE: IF NM$=NUL$ THEN NM$=
 "( UNUSED ENTRY )": NULFLG=TRUE          DP
·3100 BLK=ASC(MID$(B$,PTR+28,1))+ASC(MID$
 (B$,PTR+29,1))*256 :REM FILE LENGTH      DL
·3110 PRINT NM$;: IF NULFLG THEN PRINT: G
 OTO 3130                                 CO
·3120 PRINT ","; TYP$(TYP); ": "; FTK; FS
 C; BLK                                   BO
·3130 IF MD=192 THEN PRINT "[6" "]( FILE
 LOCKED )"                                GI
·3140 IF MD=0 AND TYP>0 THEN PRINT "> FIL
 E NOT PROPERLY CLOSED <"                 EM
·3150 RETURN                              OI
·3160 REM ------------------------------- AF
·4000 REM =======  CONTINUE  =========== DD
·4010 PRINT: PRINT "Q TO QUIT; ANY OTHER
 KEY TO CONTINUE"                         OA
·4020 GET K$: IF K$>"" THEN 4020     :REM
  EMPTY KYBD BUFFER                       LI
·4030 GET K$: IF K$="" THEN 4030          GC
·4040 PRINT SP$                           DI
·4050 IF K$="Q" THEN 4100                 JN
·4060 REM -- NEXT TRACK & SECTOR OF DIR : FM
·4070 TK=ASC(TK$+Z$): SC=ASC(SC$+Z$)      PB
·4080 IF TK=18 THEN GOTO 4110             CL
·4090 PRINT "NO MORE FILES IN DIRECTORY"  ML
·4100 CLOSE 1: CLOSE 15: END              OC
·4110 RETURN                              KM
·4120 REM ------------------------------- AJ
·5000 REM ==== SELECT FOR UNDELETION ==== KN
·5010 IF TYP>0 OR NULFLG THEN 5100        LN
·5020 PRINT "[RVSON] PRESS U TO UNDELETE
 THIS FILE[RVSOFF]"                       BE
·5030 GET K$: IF K$>"" THEN 5030:     REM
  CLEAR KYBD                              OL
·5040 GET K$: IF K$="" THEN 5040          II
·5050 PRINT SP$:IF K$<>"U" THEN GOTO 5100 GD
·5060 UDFLG=TRUE : CT=CT+1                CO
·5070 PRINT "[UP]"; TAB(35); "*"          FJ
·5080 PRINT "  (- TO BE UNDELETED -)"     IJ
·5090 B$=LEFT$(B$,PTR-1)+CHR$(128+2)+MID$
 (B$,PTR+1)                               KO
·5100 RETURN                              IN
·5110 REM ------------------------------- EI
·6000 REM ========= UNDELETE =========== CB
·6010 PRINT# 15,"B-P";2;0                 IC
·6020 PRINT# 1,TK$;SC$;B$;                BM
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 67 and 68 explain these codes and provide other essential information on entering *Ahoy!* programs. Refer to these pages **before** entering any programs!

```
·6030 GOSUB 7000                                PJ
·6040 PRINT# 15,WR$;2;DVC;TK;SC                 OF
·6050 GOSUB 7000                                AN
·6060 PRINT# 15,"V0"  :REM VALIDATE             LO
·6070 PRINT "VALIDATING DISK[3"."]"             JC
·6080 GOSUB 7000                                CL
·6090 PRINT CT; "FILE(S) UNDELETED"             BD
·6100 CLOSE 1: CLOSE 15: END                    KH
·6110 REM ---------------------------------     DE
·7000 REM ======  CHECK DISK STATUS  ====       BL
·7010 IF DS$="" THEN GOTO 7050  :REM (FOR
 C-64 ONLY)                                     PL
·7020 IF DS<20 THEN RETURN                      OC
·7030 PRINT "DISK ERROR :"; DS$ : CLOSE 1
 : CLOSE 15: END                                JF
·7040 REM === C-64 ONLY ===                     NL
·7050 INPUT# 15,EN,EM$,ET,ES                    LM
·7060 IF EN<20 THEN RETURN                      CI
·7070 PRINT "DISK ERROR :"; EN;EM$;ET;ES        AE
·7080 CLOSE 1: CLOSE 15: END                    IP
·7090 REM ---------------------------------     MM
```

MOVIE SCROLL
FROM PAGE 30

BASIC LOADER

```
·10 SA=49152                                    FG
·15 PRINT"[CLEAR]LOADING MACHINE LANGUAGE
 [3"."]"                                        GF
·20 CK=0:FORI=SA TO SA+251:READ A:POKEI,A
 :CK=CK+A:NEXTI                                 LM
·30 IF CK<>33715 THEN PRINT"DATA ERROR.":
 STOP                                           IL
·40 PRINT"USE SYS"SA"TO ACTIVATE.":END          CK
·60300 DATA 165,250,160,0,141,134               HA
·60310 DATA 2,153,0,216,153,0                   BN
·60320 DATA 217,153,0,218,153,0                 OI
·60330 DATA 219,200,208,241,165,71              HE
·60340 DATA 133,253,141,54,3,165                FF
·60350 DATA 72,133,254,141,55,3                 MJ
·60360 DATA 173,17,208,41,248,141               GO
·60370 DATA 17,208,160,0,177,253                NM
·60380 DATA 153,250,0,200,192,3                 AO
·60390 DATA 144,246,165,250,208,31              IP
·60400 DATA 173,54,3,133,253,173                MP
·60410 DATA 55,3,133,254,24,144                 KD
·60420 DATA 227,169,32,160,0,153                PB
·60430 DATA 192,7,200,192,40,144                LD
·60440 DATA 248,169,27,141,17,208               MA
·60450 DATA 96,169,32,160,0,153                 AE
·60460 DATA 192,7,200,192,40,144                LJ
·60470 DATA 248,162,24,160,0,24                 BG
·60480 DATA 32,240,255,160,0,177                MO
·60490 DATA 251,32,210,255,200,198              EE
·60500 DATA 250,208,246,160,7,132               FA
·60510 DATA 250,173,17,208,41,240               DI
·60520 DATA 24,101,250,141,17,208               IK
·60530 DATA 173,52,3,133,251,56                 FL
·60540 DATA 144,154,162,10,202,208              PF
·60550 DATA 253,198,251,208,249,173             AM
·60560 DATA 18,208,208,251,136,192              FJ
·60570 DATA 255,208,216,173,17,208              EC
·60580 DATA 41,248,24,105,7,141                 PG
·60590 DATA 17,208,165,253,24,105               ID
·60600 DATA 3,133,253,144,2,230                 JI
·60610 DATA 254,32,228,255,240,8                KF
·60620 DATA 133,2,169,27,141,17                 HK
·60630 DATA 208,96,160,0,185,40                 CA
·60640 DATA 4,153,0,4,200,208                   KK
·60650 DATA 247,185,40,5,153,0                  NP
·60660 DATA 5,200,208,247,185,40                EP
·60670 DATA 6,153,0,6,200,208                   NE
·60680 DATA 247,185,40,7,153,0                  AC
·60690 DATA 7,200,208,247,172,53                PC
·60700 DATA 3,162,10,202,208,253                HD
·60710 DATA 136,208,248,24,144,148              OJ
```

DEMO

```
·10 PRINT"[CLEAR]LOADING MACHINE LANGUAGE
 [3"."]"                                        GK
·20 DIM A$(30):POKE53280,0:POKE53281,0:PR
 INT"[CLEAR]"                                   FE
·30 FORI=1TO30:READA$(I):NEXT                   EL
·40 POKE820,45:POKE821,200:REM --- SCROLL
 SPEED ---                                      NI
·50 POKE250,3:REM --- TEXT COLOR ---            KL
·60 REM --- A$(20)="" MEANS LAST STRING -
 --                                             GI
·70 A$(1)=MID$(A$(1),1):SYS49152                OG
·80 END                                         NA
·90 REM --- TEXT TO SCROLL ---                  LN
·100 DATA " "," "," ","THIS IS MOVIE SCRO
 LL[3"."]"," "                                  NO
·110 DATA "THE FINE-SCROLL UTILITY WHICH
 YOU HAVE","BEEN WAITING FOR[3"!"]"             KE
·120 DATA " ","SCROLL YOUR HELP SCREENS[3
 "."]"," ","OR YOUR TITLE SCREENS[3"."]"        JG
·130 DATA " ","OR USE IT TO TITLE YOUR HO
 ME VIDEOS!"                                    ED
·140 DATA " "," "," ","[3"*"] ALL WITH [R
 VSON]MOVIE SCROLL[3"!"][RVSOFF] [3"*"]",
 " "," "                                        GA
·150 DATA "IT'S SAFE, NON-TOXIC, AND HARM
 LESS"                                          EM
·160 DATA "AS WELL AS SIMPLE AND CONVENIE
 NT."                                           IK
·170 DATA "ALL YOU DO IS SET UP A STRING
 ARRAY!"                                        CA
·180 DATA "NOW, UNLESS YOU WANT TO SIT HE
 RE ALL"                                        EN
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 67 and 68 explain these codes and provide other essential information on entering ***Ahoy!*** programs. Refer to these pages **before** entering any programs!

```
•190 DATA "DAY LONG WATCHING THIS DEMO,HI
 T ANY KEY"                                 OD
•200 DATA "TO RETURN CONTROL TO BASIC[4".
 "]"," "," "," "                            KE
•210 DATA "AH WELL, HERE WE GO AGAIN[3"."
 ]",""                                      AC
```

PROGRAMMING YOUR OWN TEXT GAMES

FROM PAGE 39

LOST DUTCHMAN'S MINE

```
•10 REM LOST DUTCHMAN'S MINE                BH
•20 REM BY CLEVELAND M. BLAKEMORE           PC
•30 REM RUNS ON ALL COMMODORE COMPUTERS     EH
•40 REM WITH 40 COLUMNS AND PETASCII        IL
•50 REM LOCK IN UPPERCASE & CLEAR SCREEN    HF
•60 PRINTCHR$(8)CHR$(142)CHR$(147)          KA
•70 REM PRINT TITLE SCREEN                  JA
•80 PRINT"[7"[DOWN]"]"TAB(10)"LOST DUTCHM
 AN'S MINE"                                 LF
•90 PRINTTAB(10)"[DOWN][DOWN]* ALL C= COM
 PUTERS *":PRINTTAB(14)"[DOWN](C)1988 CMB
 "                                          JA
•100 PRINTTAB(8)"[DOWN][DOWN]BY CLEVELAND
  M. BLAKEMORE"                             NJ
•110 REM INITIALIZE ARRAYS AND VARIABLES    NP
•120 DIM V$(32),N$(20),L$(9),S$(9),D(9,6)
 ,O(9,8),I(8),D$(8),M$(4)                   CA
•130 REM VARIABLES AND FLAGS                MA
•140 REM CL=CURRENT LOCATION                KA
•150 DIM CL,WINGAME,WEIGHT,DEAD,ALIVE,WOL
 F,GHOST,FULL,EMPTY,RIVER,MOUND,COUNT       FB
•160 ALIVE=1:DEAD=0:WOLF=ALIVE:SPIRIT=ALI
 VE                                         NJ
•170 FULL=1:EMPTY=0:RIVER=EMPTY:MOUND=FUL
 L:COUNT=0                                  IP
•180 REM READ LOCATIONS & DESCRIPTIONS      HP
•190 FORX=.TO9:READL$(X):NEXT:FORX=.TO9:R
 EADS$(X):NEXT                              PM
•200 REM READ IN DIRECTIONS                 LE
•210 FORX=1TO9:FORD=1TO6:READD(X,D):NEXTD
 ,X                                         NL
•220 FORX=1TO32:READV$(X):NEXT:FORX=1TO20
 :READN$(X):NEXT                            DD
•230 FORX=.TO8:READD$(X):NEXT               BI
•240 READX,D:IFX=-1THEN260                  AN
•250 O(X,D)=1:GOTO240                       MM
•260 FORX=.TO4:READM$(X):NEXT               FO
•270 REM WAIT FOR RETURN KEY PRESS          AM
•280 PRINTTAB(8)"[6"[DOWN]"]PRESS RETURN
 TO CONTINUE"                               JG
•290 GETA$:ON-(A$<>CHR$(13))GOTO290         DN
•300 REM START OFF AT GAS STATION           KH
•310 CL=.:GOSUB730                          PL
•320 REM PARSER MAIN ROUTINES               FK
•330 REM GET COMMAND                        KK
•340 IN$="":V=.:N=.                         OE
•350 INPUT"[DOWN]COMMAND>";IN$:IFIN$=""TH
 ENPRINT"SILENCE IS GOLDEN.":GOTO350        JK
•360 REM PARSE VERB                         KE
•370 IFLEN(IN$)>1THEN450                    OD
•380 V=.:FORX=1TO7:IFIN$=V$(X)THENV=X       AK
•390 NEXT:IFV=.THENPRINT"WHAT DOES THAT M
 EAN?":GOTO600                              DK
•400 IFV=7THENGOSUB880:GOTO600              IC
•410 ON-(CL=6AND(V=1ORV=6)ANDWEIGHT>1)GOT
 O420:GOTO430                               EN
•420 PRINT"YOU ARE CARRYING TOO MUCH TO F
 IT THROUGHTHE CRACK.":GOTO600              HM
•430 IFD(CL,V)THENCL=D(CL,V):GOSUB730:GOT
 O600                                       OA
•440 PRINT"YOU CAN'T GO THAT WAY!":GOTO60
 0                                          NK
•450 V=.:FORX=8TO13:IFIN$=V$(X)THENV=X      GA
•460 NEXT:IFVTHENV=V-7:GOTO410              NA
•470 V=.:FORX=14TO32:IFLEFT$(IN$,LEN(V$(X
 )))=V$(X)THENV=X:D=LEN(V$(X))              EB
•480 NEXT:IFV=.THENPRINT"I DON'T KNOW THA
 T VERB!":GOTO600                           HA
•490 REM PARSE NOUN                         CJ
•500 X=LEN(IN$):IFX=DTHEN570                GK
•510 IFMID$(IN$,D,1)<>CHR$(32)ANDD<XTHEND
 =D+1:GOTO510                               IP
•520 IFD=XTHEN570                           CH
•530 D=D+1:N=.:FORX=1TO20:IFMID$(IN$,D,LE
 N(N$(X)))=N$(X)THENN=X                     PB
•540 NEXT:IFN=.THENPRINTV$(V)" WHAT?!?":G
 OTO600                                     IJ
•550 REM JUMP TO APPROPRIATE SUBROUTINE     DI
•560 REM BASED ON VALUE IN "V"              CB
•570 IFV<23THENONV-13GOSUB880,880,920,970
 ,970,1030,1030,1100,1160:GOTO600           AO
•580 ONV-22GOSUB1210,1210,1280,1310,1310,
 1310,1370,1370,1450,1450                   PE
•590 REM CHECK FLAGS AND COUNTERS           KA
•600 IFWINGAMETHENX=.:GOTO700               CK
•610 IFCL=3ANDSPIRIT=ALIVETHENCL=4:GOSUB7
 30:PRINT"[DOWN]THE SPIRIT SCARES YOU BAC
 K!!"                                       HO
•620 IFCL=9ANDRIVER=EMPTYTHENCL=7:GOSUB73
 0:PRINT"[DOWN]THE ROOM BELOW IS UNDERWAT
 ER."                                       CH
•630 IFRIVER=FULLANDCL=8ANDCOUNT>0THENX=1
 :GOTO700                                   IN
•640 IFWOLF=ALIVEANDCL=7ANDCOUNT>1THENX=2
 :GOTO700                                   PG
•650 IFSPIRIT=ALIVEANDCL=4ANDCOUNT>4THENX
 =3:GOTO700                                 NH
•660 IFSPIRIT=ALIVEANDCL=4ANDCOUNT>2THENP
```

```
RINT"[DOWN]THE SPIRIT LOOKS REAL ANGRY!" FC
·670 IF((CL>1ANDCL<4)ORCL>5)ANDI(0)>-1AND
COUNT>2ANDRND(0)>.5THENX=4:GOTO700       JA
·680 COUNT=COUNT+1                        AF
·690 GOTO340                              NH
·700 PRINT:PRINTM$(X):IFXTHENPRINT"SORRY,
 YOU HAVE DIED[3"."]"                     GJ
·710 END                                  EI
·720 REM PRINT LOCATION                   FD
·730 COUNT=0                              FG
·740 PRINTCHR$(147);                      FP
·750 IF((CL>1ANDCL<4)ORCL>5)ANDI(0)>-1THE
NPRINT"IT'S PITCH DARK IN HERE!":RETURN  EL
·760 PRINT"YOU ARE":PRINTL$(CL)           OP
·770 REM PRINT AVAILABLE DIRECTIONS       LG
·780 PRINT"DIRECTIONS:";                  GI
·790 D=.:FORX=1TO6:IFD(CL,X)THENPRINTV$(7
+X);",";:D=X                              AK
·800 NEXT:IFD=.THENPRINT"NONE."           IP
·810 IFDTHENPRINT"[LEFT]."                EE
·820 REM PRINT OBJECTS AT LOCATION        BC
·830 PRINT"YOU CAN NOW SEE:":IFS$(CL)>""T
HENPRINTS$(CL)                            GG
·840 D=.:FORX=.TO8:IFO(CL,X)THENPRINTD$(X
):D=1                                     FC
·850 NEXT:IFD=.ANDS$(CL)=""THENPRINT"NOTH
ING."                                     DM
·860 RETURN                               ON
·870 REM LOAD/INVENTORY                   CM
·880 PRINT"YOU ARE CARRYING:":D=.:FORX=.T
O8:IFI(X)THENPRINTD$(X):D=1               KJ
·890 NEXT:IFD=.THENPRINT"NOTHING."        IL
·900 RETURN                               BF
·910 REM GOZER                            JF
·920 IFCL=.THENCL=1:GOTO950               AC
·930 IFCL=1THENCL=.:GOTO950               CA
·940 PRINT"NOTHING HAPPENS HERE.":RETURN  EM
·950 GOSUB730:RETURN                      KE
·960 REM TAKE/GET                         IA
·970 IFN=.THENPRINT"GET WHAT?!?":RETURN   LL
·980 IFN>9THENPRINT"YOU CAN'T GET THAT!":
RETURN                                    NG
·990 IFI(N-1)THENPRINT"YOU ALREADY HAVE I
T!":RETURN                                HN
·1000 IFO(CL,N-1)THENI(N-1)=O(CL,N-1):O(C
L,N-1)=.:PRINT"OK,TAKEN.":GOTO1070        MN
·1010 PRINT"IT'S NOT HERE!":RETURN        BI
·1020 REM DROP/PUT                        OL
·1030 IFN>9THENPRINT"YOU MUST BE CONFUSED
.":RETURN                                 KE
·1040 IFI(N-1)=.THENPRINT"YOU DON'T HAVE
IT!":RETURN                               HK
·1050 O(CL,N-1)=I(N-1):I(N-1)=0:PRINT"OKA
Y, DROPPED."                              FP
·1060 IFO(0,5)ANDO(0,7)ANDO(0,8)THENWINGA
ME=1                                      PC
·1070 WEIGHT=0:FORX=.TO8:IFI(X)THENWEIGHT
=WEIGHT+1                                 DC
·1080 NEXT:RETURN                         ID
·1090 REM LIGHT                           EH
·1100 IFI(1)=.THENPRINT"NO MATCHES.":RETU
RN                                        PF
·1110 IFN<>1THENPRINT"YOU'RE A PYROMANIAC
!":RETURN                                 NN
·1120 IFI(.)=.THENPRINT"YOU DON'T HAVE IT
!":RETURN                                 CL
·1130 IFI(.)=-1THENPRINT"IT'S ALREADY BUR
NING.":RETURN                             MK
·1140 I(.)=-1:GOSUB730:PRINT"[DOWN]OKAY,
IT'S BURNING.":RETURN                     JH
·1150 REM DIG                             OI
·1160 IFCL<>9ORN<>15THENPRINT"TRY IT SOME
WHERE ELSE.":RETURN                       BM
·1170 IFMOUND=EMPTYTHENPRINT"THE MOUND HA
S ALREADY BEEN EXCAVATED.":RETURN         KB
·1180 O(9,8)=1:MOUND=EMPTY:GOSUB730       NN
·1190 PRINT"[DOWN]OKAY, YOU HAVE DUG A DE
EP HOLE[3"."]":RETURN                     AD
·1200 REM SHOOT/KILL                      BG
·1210 IFI(3)=.THENPRINT"YOU DON'T HAVE TH
E "N$(4)"VER.":RETURN                     ID
·1220 IFI(6)=.THENPRINT"YOU DON'T HAVE TH
E "N$(7)"TS.":RETURN                      MC
·1230 IFN<>13ANDN<>14THENPRINT"YOU'RE CRA
ZY.":RETURN                               DM
·1240 IFCL<>7THENPRINT"IT'S NOT HERE.":RE
TURN                                      GN
·1250 WOLF=DEAD:PRINT"[DOWN]THE REVOLVER
THUNDERS FIRE AND THE WOLF FALLS OVER DE
AD";                                      DG
·1260 PRINT"[RIGHT]IN MID-LEAP.":S$(7)="A
 DEAD TIMBER WOLF":RETURN                 OB
·1270 REM SAY                             HF
·1280 IFN=10THEN920                       CP
·1290 PRINT"NOTHING HAPPENS.":RETURN      DG
·1300 REM TURN/ROTATE/TWIST               HL
·1310 IFCL<>8ORN<>18THENPRINT"INTERESTING
 CONCEPT[3"."]":RETURN                    NH
·1320 RIVER=FULL:PRINT"THE DOOR BLASTS OP
EN WITH A TIDAL WAVE  OF WATER[3"!"]"     MC
·1330 PRINT"THE RIVER BED QUICKLY FILLS U
P AND YOU[RIGHT][RIGHT]ARE DRENCHED."     MN
·1340 PRINT"YOU'D BETTER LEAVE. IT'S ALMO
ST NECK[4"[RIGHT]"]DEEP IN HERE!"         DK
·1350 S$(8)="BUBBLES COMING OUT OF YOUR C
LENCHED LIPSUNDERWATER":COUNT=0:RETURN    LK
·1360 REM BLOW/WHIST                      NF
·1370 IFN=.THENPRINT"TWEET TWEET TWEET[3"
."]"                                      AC
·1380 IFCL=4ANDN=.THENPRINT"THE SPIRIT TR
EMBLES AND WAVERS A LITTLE.":RETURN       BB
·1390 IFN<>5THENPRINT"THAT'S RIDICULOUS!"
:RETURN                                   BC
·1400 IFI(N-1)=.THENPRINT"YOU DON'T HAVE
IT!":RETURN                               NH
·1410 IFCL<>4THENPRINT"NOTHING HAPPENS.":
```

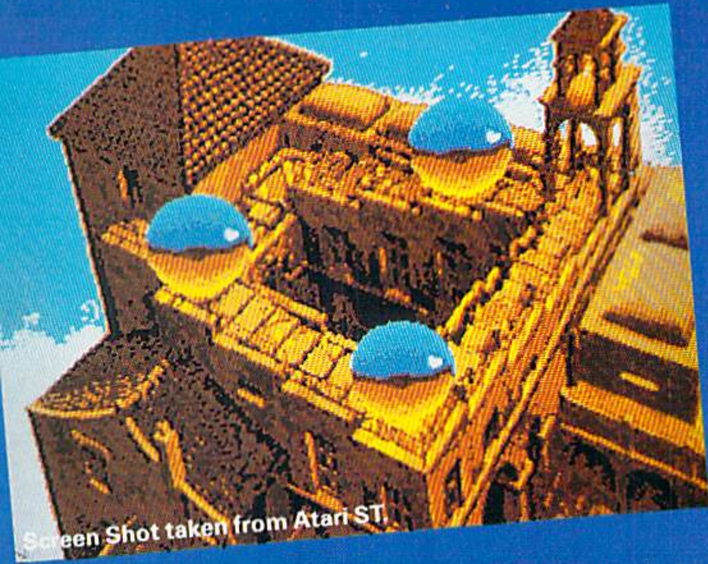
```
RETURN                                      EI
·1420 SPIRIT=DEAD:PRINT"THE SPIRIT WRITHE
S AND VANISHES IN A[4"[RIGHT]"]CLOUD OF
SMOKE!"                                     ED
·1430 S$(4)="":RETURN                       DJ
·1440 REM LOOK/EXAMINE                      IP
·1450 IFN=.THENGOSUB740:RETURN              AE
·1460 IFN<10THENIFI(N-1)=.THENPRINT"YOU D
ON'T HAVE IT!":RETURN                       PC
·1470 IFN=2THENPRINT"GOZER TRAVEL INC.'NE
ED TO TRAVEL?? CALL GOZER!!'":RETURN        OL
·1480 IFCL=2ANDN=12THENPRINT"HIS BONY CLA
W POINTS TO THE EAST[3"."]":RETURN          KP
·1490 IFCL=0ANDN=19THENPRINT"[RVSON] PUT
ALL THE TREASURE ON THIS SHELF ":RETURN     BD
·1500 IFN=1ANDI(0)=-1THENPRINT"IT'S BURNI
NG.":RETURN                                 DF
·1510 IFN=1ANDI(0)=1THENPRINT"IT'S OUT.":
RETURN                                      OM
·1520 PRINT"YOU SEE NOTHING OUT OF THE OR
DINARY.":RETURN                             LB
·1530 REM LOCATIONS                         CI
·1540 DATA IN A DESERTED LAST CHANCE GAS
STATION                                     FP
·1550 DATA"IN A DUSTY ADOBE HOUSE WITH A
SUNBEAM[3"[RIGHT]"]COMING THROUGH AN EAS
T WINDOW"                                   FP
·1560 DATA IN A LONG E-W MINESHAFT,IN A S
LOPING GRANITE TUNNEL                       KC
·1570 DATA IN FRONT OF AN OLD BOARDED UP
MINESHAFT ENTRANCE,IN A DEEP RAVINE         OH
·1580 DATA INSIDE A COOL CAVERN WITH A SM
ALL CRACK LEADING DOWN TO THE NORTH         BF
·1590 DATA ON A SPIRAL STONE STAIRCASE      LL
·1600 DATA IN A SUBTERRANEAN RIVER BED RU
NNING EAST & WEST                           DM
·1610 DATA IN AN ANCIENT INDIAN BURIAL GR
OUND                                        CK
·1620 REM SPECIFIC OBSERVATIONS             MA
·1630 DATA A METAL PLAQUE ABOVE A GLASS S
HELF,A SKULL MOUNTED ON A SPEAR             AJ
·1640 DATA A SKELETON DRAPED WITH COBWEBS
,"",A WHISTLING EVIL SPIRIT                 AG
·1650 DATA "",A COLOSSAL STONE SUN CALEND
AR,A RAVENOUS SNARLING TIMBER WOLF!         AD
·1660 DATA A CIRCULAR WHEEL SET IN THE MI
DDLE OF ANIRON DOOR TO THE EAST             AK
·1670 DATA A LARGE BURIAL MOUND             BL
·1680 REM DIRECTIONS                        AI
·1690 REM  N,S,E,W,U,D                      NM
·1700 DATA 4,0,0,0,0,0                      CA
·1710 DATA 0,0,6,7,3,0                      FE
·1720 DATA 0,0,0,0,4,2                      FC
·1730 DATA 3,1,0,5,0,0                      GL
·1740 DATA 0,0,4,0,0,0                      EA
·1750 DATA 8,0,0,2,0,8                      HA
·1760 DATA 0,0,2,0,2,9                      HD
·1770 DATA 0,6,0,8,6,0                      JG
·1780 DATA 0,0,0,0,7,0                      GD
·1790 REM VERBS                             CN
·1800 DATA N,S,E,W,U,D,I,NORTH,SOUTH,EAST
,WEST,UP,DOWN,LOAD,INVEN,GOZER,TAKE         MC
·1810 DATA GET,DROP,PUT,LIGHT,DIG,SHOOT,K
ILL,SAY,TURN,ROTAT,TWIST,BLOW,WHIST         GH
·1820 DATA LOOK,EXAMINE                     CE
·1830 REM NOUNS                             EL
·1840 DATA TORCH,MATCH,SHOVE,REVOL,BOTTL,
DIAMO,BULLE,RUBY,NUGGE,GOZER                HF
·1850 DATA SKULL,SKELE,WOLF,TIMB,MOUND,CA
LEN,SPIRI,WHEEL,PLAQU,SHELF                 PE
·1860 REM DESCRIPTIONS                      IN
·1870 DATA A WOODEN TORCH,A MATCHBOOK,A R
USTY SHOVEL                                 HF
·1880 DATA A PEARL HANDLED REVOLVER,AN EM
PTY 7-UP BOTTLE                             DK
·1890 DATA A GLOWING DIAMOND,A HANDFUL OF
 RUSTY BULLETS,A LUSTROUS RED RUBY          FG
·1900 DATA THE DUTCHMAN'S NUGGET            JC
·1910 REM OBJECTS & INITIAL LOCATIONS       NM
·1920 DATA 0,1,0,3,0,4,1,0,5,2,2,6,3,7,8,
5,-1,-1                                     DD
·1930 REM END OF GAME MESSAGES              PA
·1940 DATA CONGRATULATIONS! YOU'VE WON TH
E GAME!                                     KL
·1950 DATA YOU DROWNED IN THE RIVER.,THE
TIMBER WOLF TEARS YOU TO SHREDS.            BK
·1960 DATA THE EVIL SPIRIT SUCKED THE BRE
ATH OUT OF YOU!                             GJ
·1970 DATA YOU FELL DOWN IN THE DARK AND
BROKE YOUR NECK!                            GB
```

...COMING IN THE JULY ISSUE OF *AHOY!* (ON SALE JUNE 7)...

THERMAL CONNECTION
ANOTHER REAL-WORLD INTERFACE

TOMB OF HORROR
LAST IN THE 3-D BLAKEMORE TRILOGY

THINK MODULES!
TEXT GAME PROGRAMMING, PART II



Screen Shot taken from Atari ST.



Screen Shot taken from Atari ST.



Screen Shot taken from C-64

Firebird are running an Advanced Art Studio promotion with Ahoy! Magazine, Atari Explorer Magazine, and Quantumlink. Look in Ahoy!, Atari Explorer or on Quantumlink for details, or clip this coupon and mail it to AAS Competition, Firebird Licensees, Inc. 71 Franklin Turnpike, Waldwick, NJ. 07463 for details.

Name ....

Address ....

....

.... Age ....

* No purchase necessary. Recorded competition information:- (718) 847 0234

THE ADVANCED ART STUDIO

The only art package that lets you design sprites, animate, and fly to Paris for the weekend . . .

Unleash the creative potential of your Atari ST or Commodore 64. Among the stunning features you will find at your fingertips:

On the 520 *Icon driven Toolbox with all options available on one screen *2 work screens plus separate sprite design grid *High speed draw, shapes, spray, fill, copy, brush and picture manipulation *Sophisticated window routines including mask, stretch, squash, flip, smooth and rotate *All drawing functions are available in magnify mode plus directional scrolling of the magnifying grid in four directions *Extensive sprite design system including choice of sprite size, large on-screen sprite store, auto outline, draw with sprites as brushes and much more *all sprites can be out put as data ready for inclusion in other programs *On screen animation of sprites.

On the 64 *Windows, Icons-pull down menus-pointing devices *Operates in high-res mode. Full control over color attributes* All information on screen *16 pens, 8 random sprays, 16 user definable brushes* Windows can be inverted, cut and pasted, enlarged, reduced, squashed, stretched, flipped and rotated *Solid and texture fill option *Wash texture feature *3 levels of magnification with pan and zoom, pixel clear and plot *Full range of facilities in multi-color mode plus transfer from hi-res to multi-color *Comprehensive edit.

CREDIT CARD HOTLINE
(201) 444-5700

RAINBIRD FIREBIRD

"The First Full Line in Software"
Firebird Licensees, Inc.
P.O. Box 49, Ramsey, NJ 07446
(201) 444-5700



ECHELON is a true flight simulator which takes you to the edge of our solar system. Through the retrieval of artifacts and information, you must find the solution to a series of mysterious and puzzling events happening inside the area. 'Echelon' also includes 'The LipStik', Voice Activated Control Headset. Retail price is $44.95.

WORLD CLASS LEADER BOARD - Find out why 'World Class' Leader Board is the most popular golf simulation program ever! 'World Class' features 3 famous courses with outstanding graphics and realistic play. Retail price is $39.95.

FAMOUS COURSES OF THE WORLD - These disks are available for use with your 'World Class' Leader Board for $19.95.

VOL. I - Features Harbour Town, Sunningdale, Dorado Beach and Pine Ridge.

VOL. II - Features Pebble Beach, Muirfield, Colonial Country Club and Glenmoor Country Club.

VOL. III - Features Sawgrass, Royal St. Georges, Bamff Springs and Firestone Country Club.

LEADER BOARD TRIPLE PACK - Features 'Leader Board', 'Tournament Disk #1' and 'Executive Leader Board' for $19.95.

TRIPLE PACK - Features 'Beach Head', 'Raid Over Moscow', and 'Beach Head II' for the incredible price of $19.95.

MACH 128 - Load your C-64 or C-128 programs 7 times faster with this utility cartridge. Now works with the 1541, 1571 and the new 1581 disk drives. Retail price is $49.95.

LIPSTIK PLUS - 'The Voice Activated Control Headset' is used with your joystick and gives you a greater feeling of power and control. You'll find yourself more absorbed in the action than you ever thought possible. Retail price is $24.95.

ACCESS™ Software Incorporated

If you can't find our products at your local dealer, you can order direct. For mail orders, enclose check or money order, plus $1.00 shipping and handling, and specify machine version desired. Order by phone on VISA, MasterCard or C.O.D. by calling TOLL FREE (800) 824-2549.
ACCESS SOFTWARE, INC. 545 W. 550 S., Suite 130, Bountiful, UT 84010